FIGURE 1

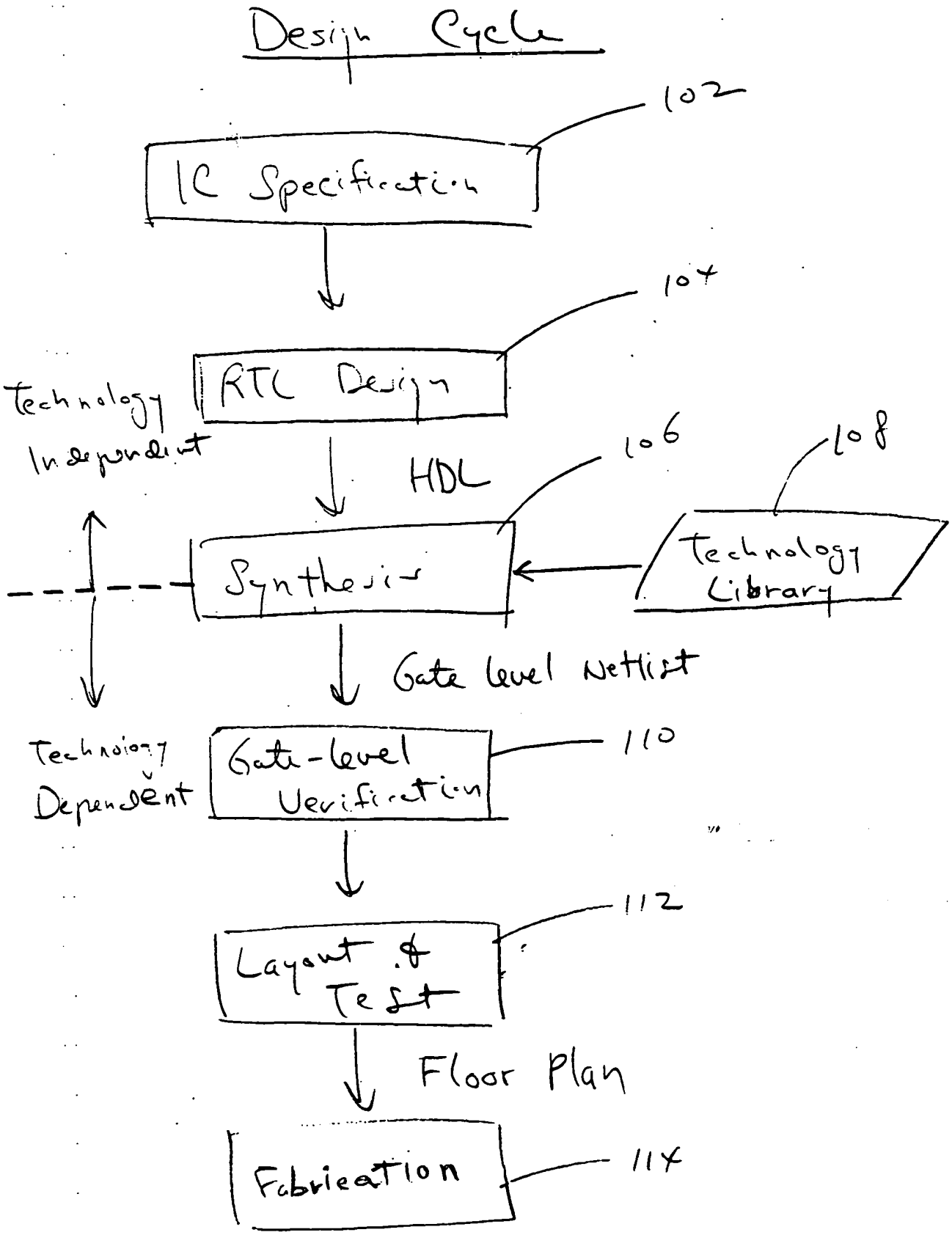

Design Cycle
IC Specification
102
RTL Design
104
Technology Independent
HDL
Synthesis
106
Technology Library
108
Gate level Netlist
Technology Dependent
Gate-level Verification
110
Layout & Test
112
Floor Plan
Fabrication
114

FIGURE 2

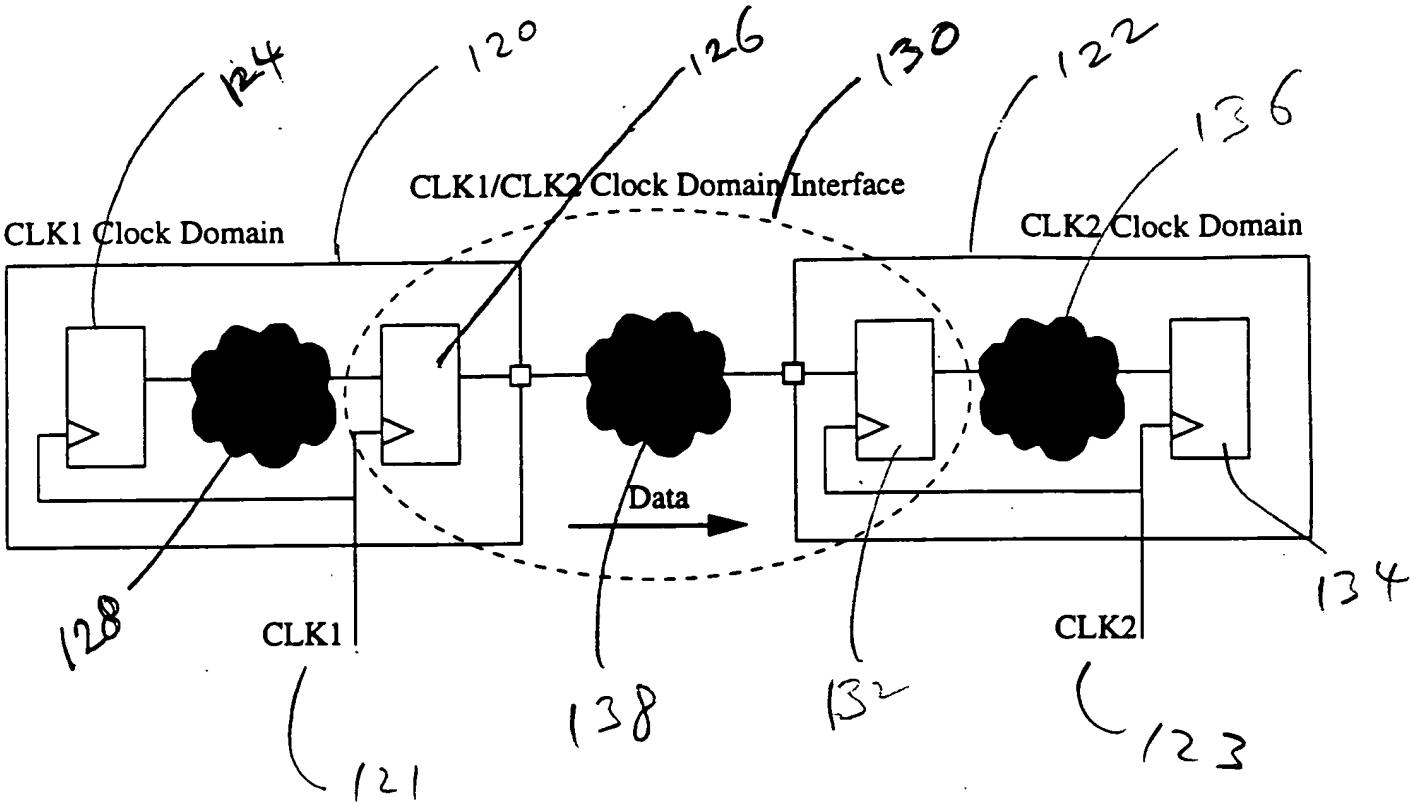


FIGURE 4

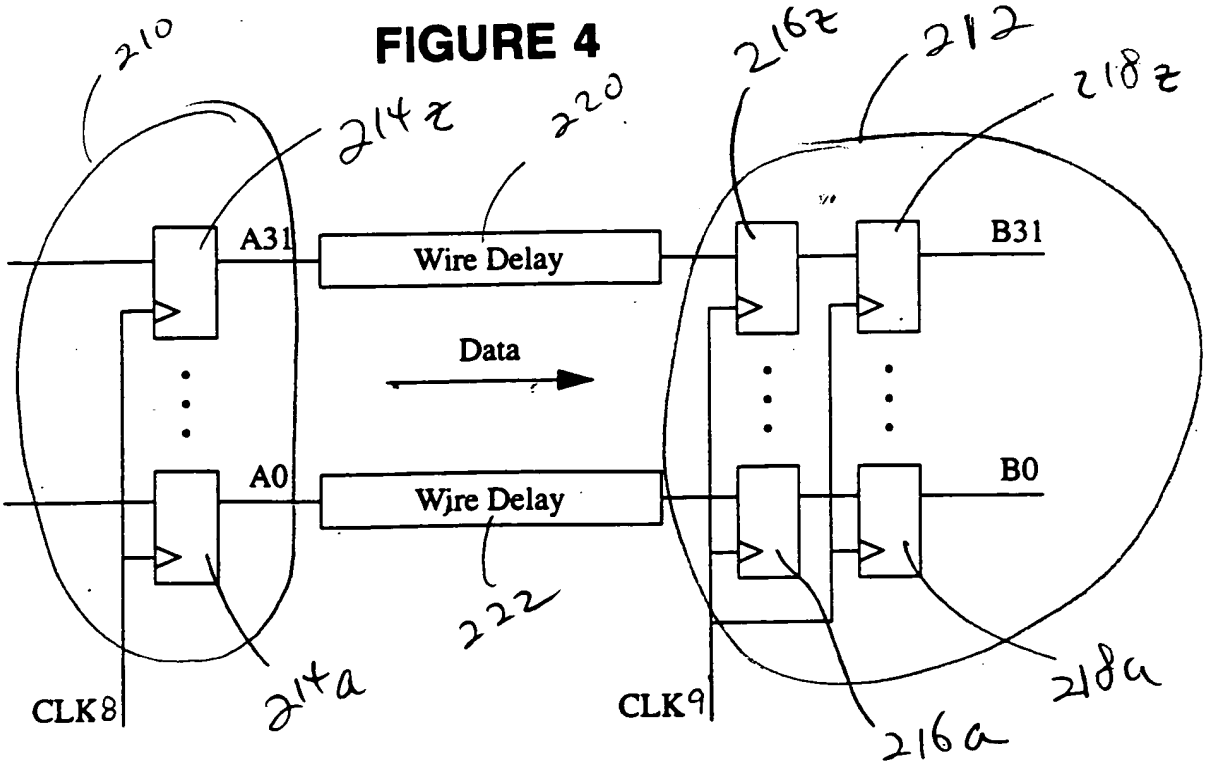
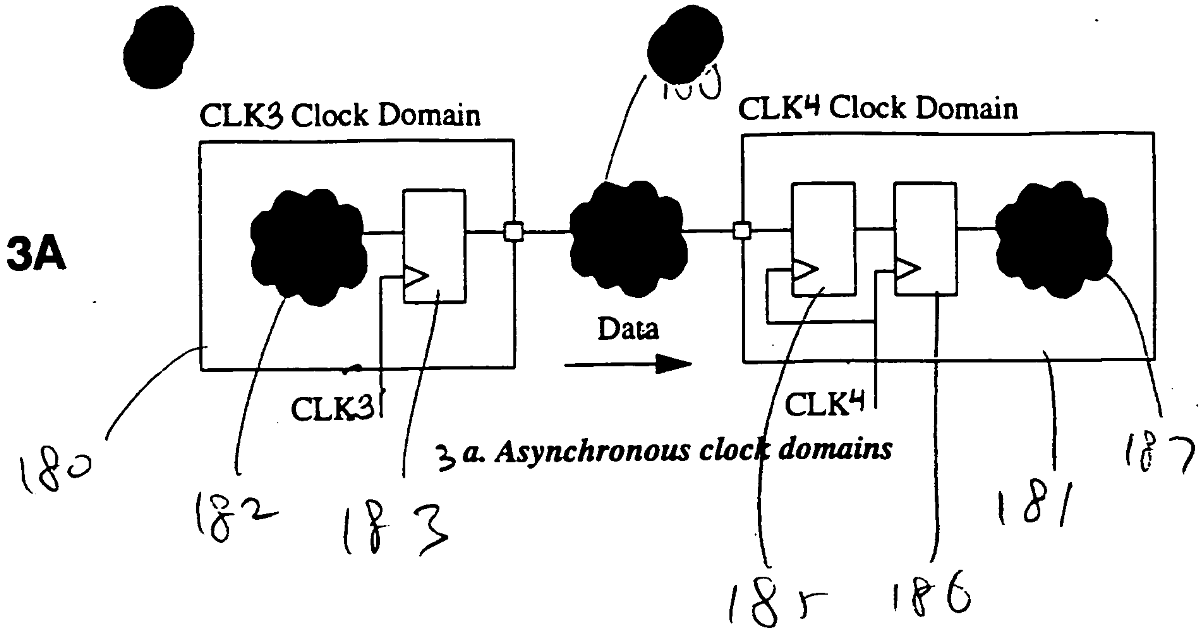
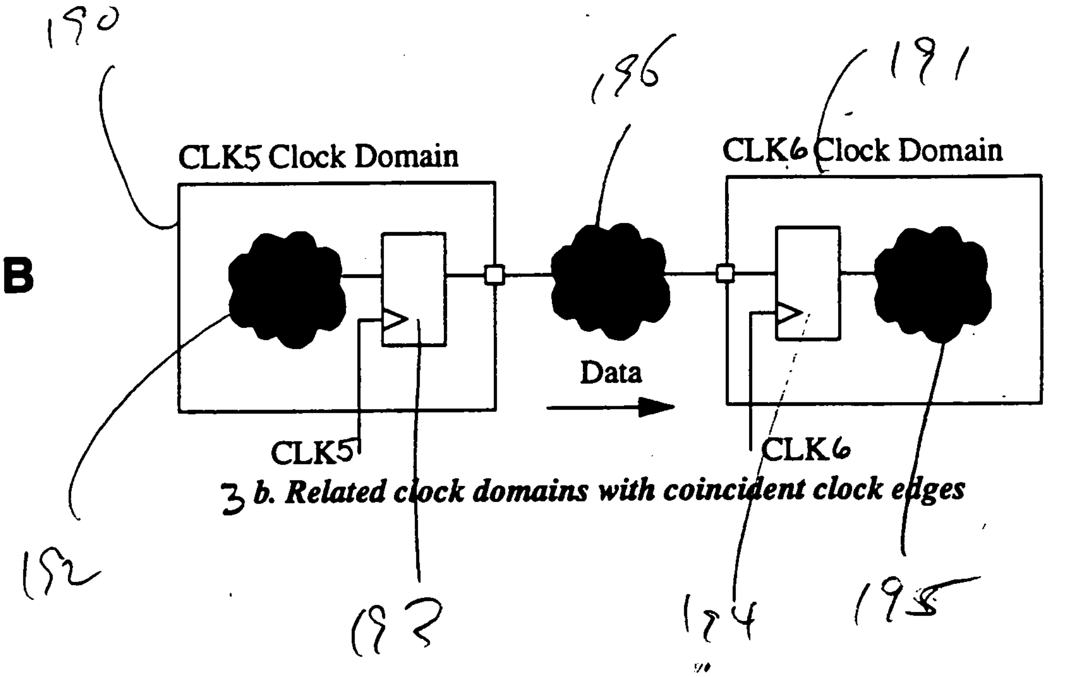


FIGURE 3A



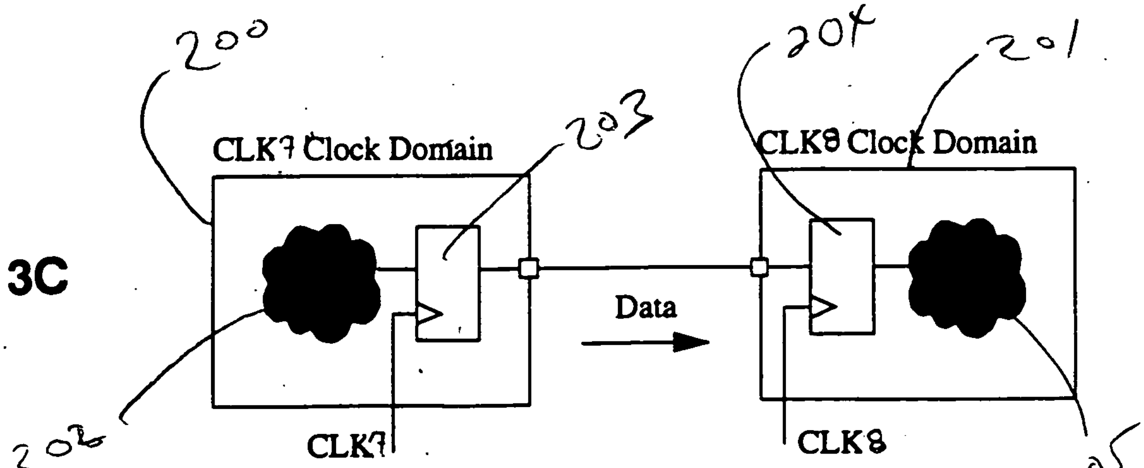
3 *a. Asynchronous clock domains*

FIGURE 3B



3 *b. Related clock domains with coincident clock edges*

FIGURE 3C



3 *c. Related clock domains with sequenced clock edges*

FIGURE 5

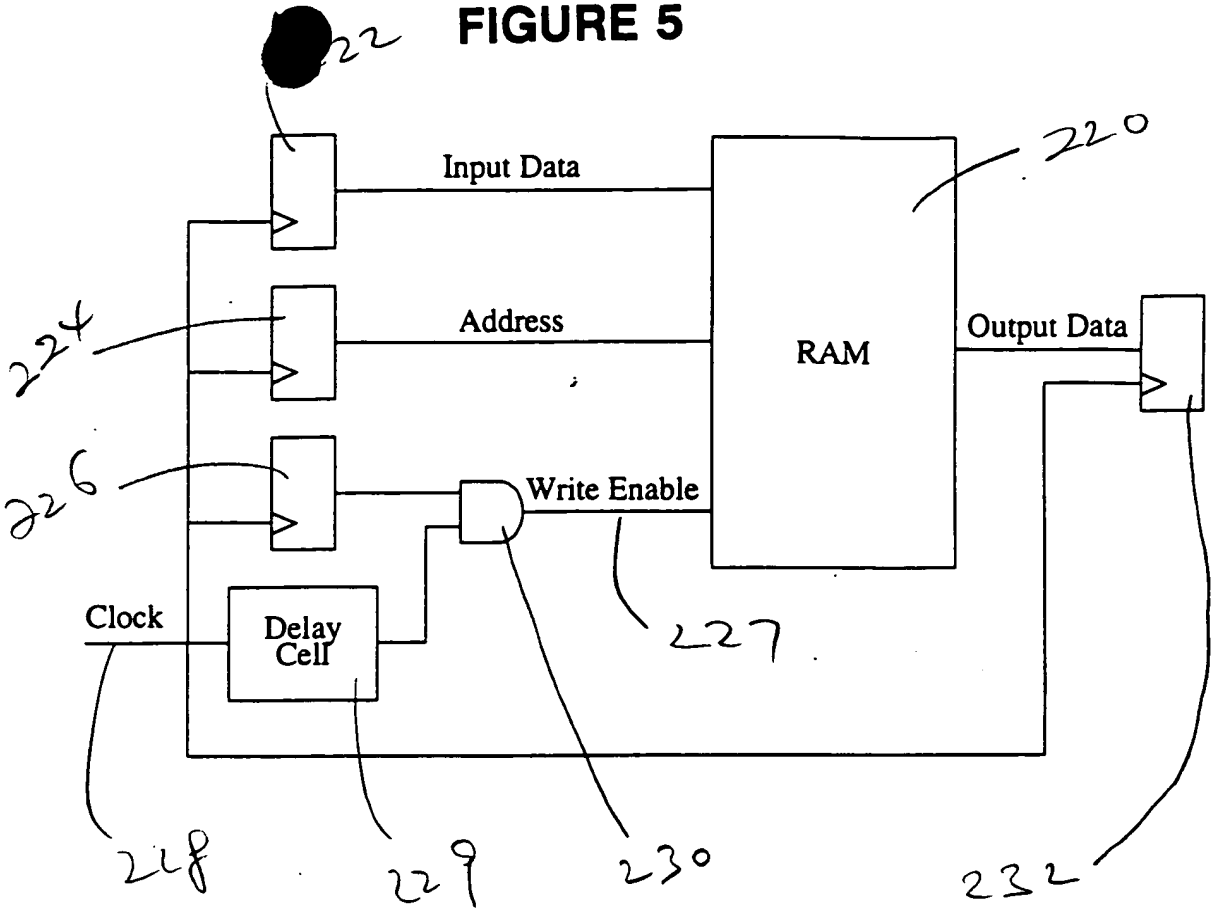
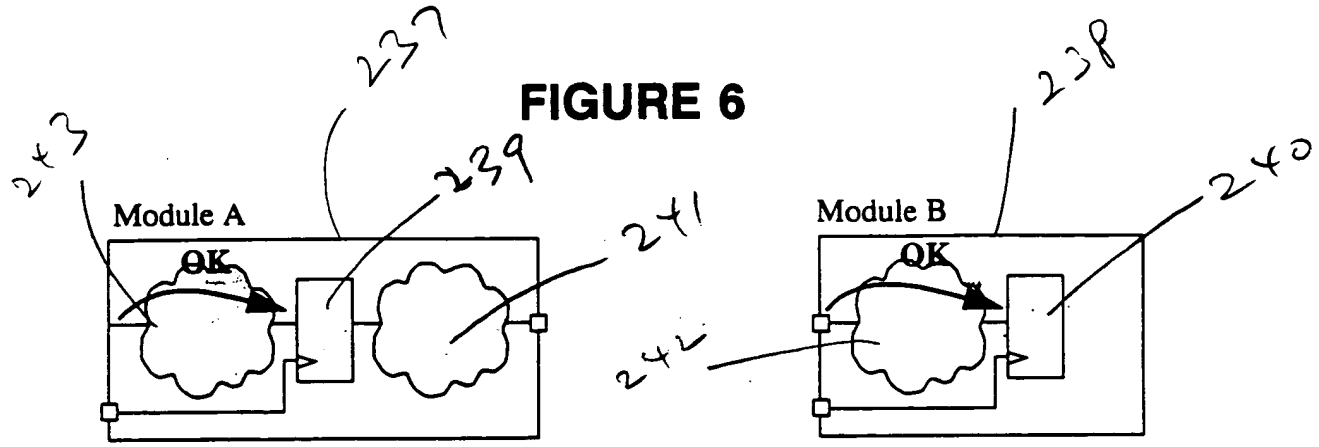
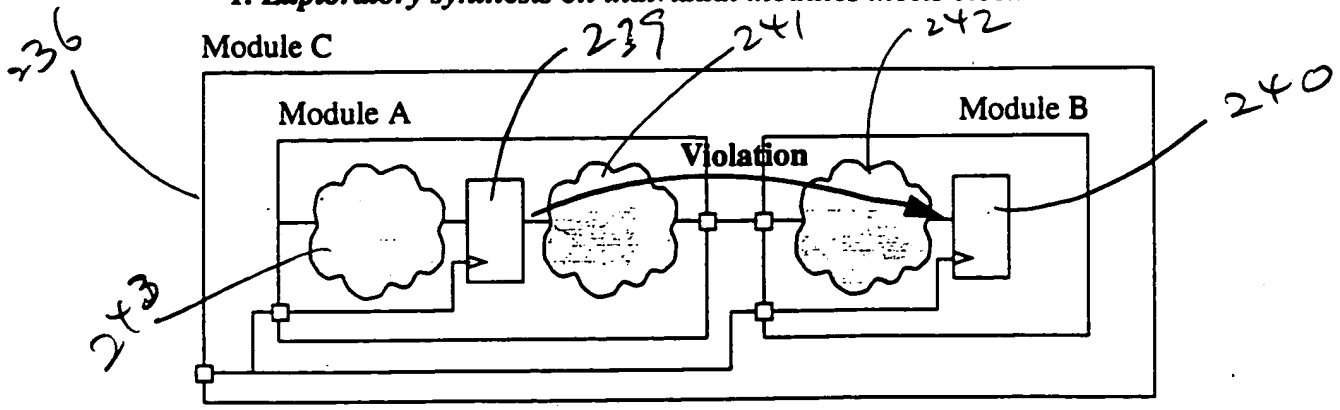


FIGURE 6



1. Exploratory synthesis on individual modules meets clock constraints.



2. Violating paths show up when modules are put together.

FIGURE 7

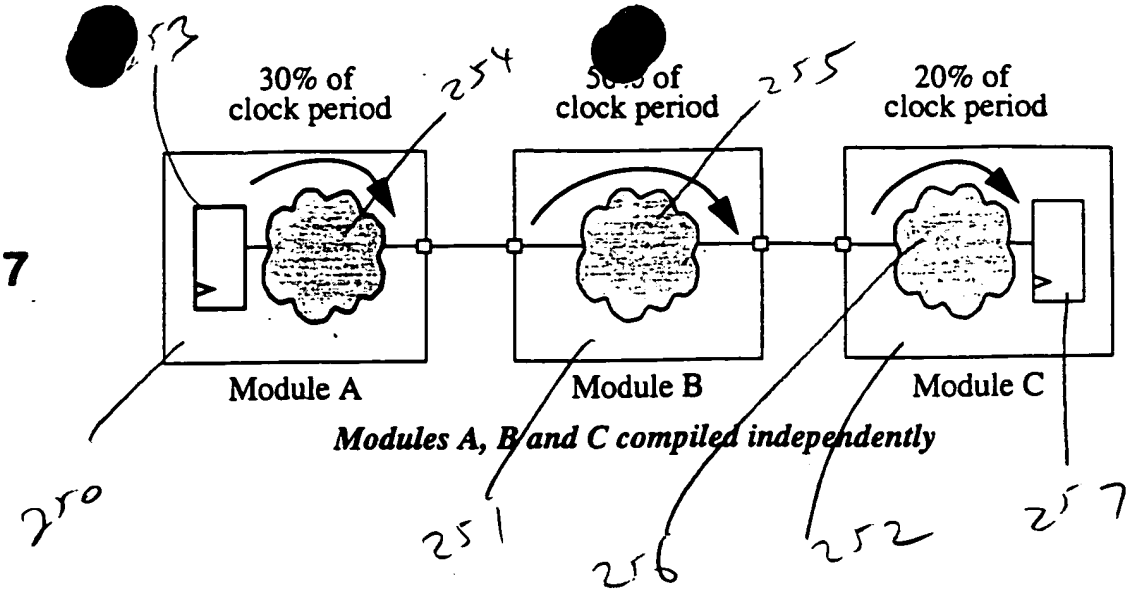


FIGURE 8

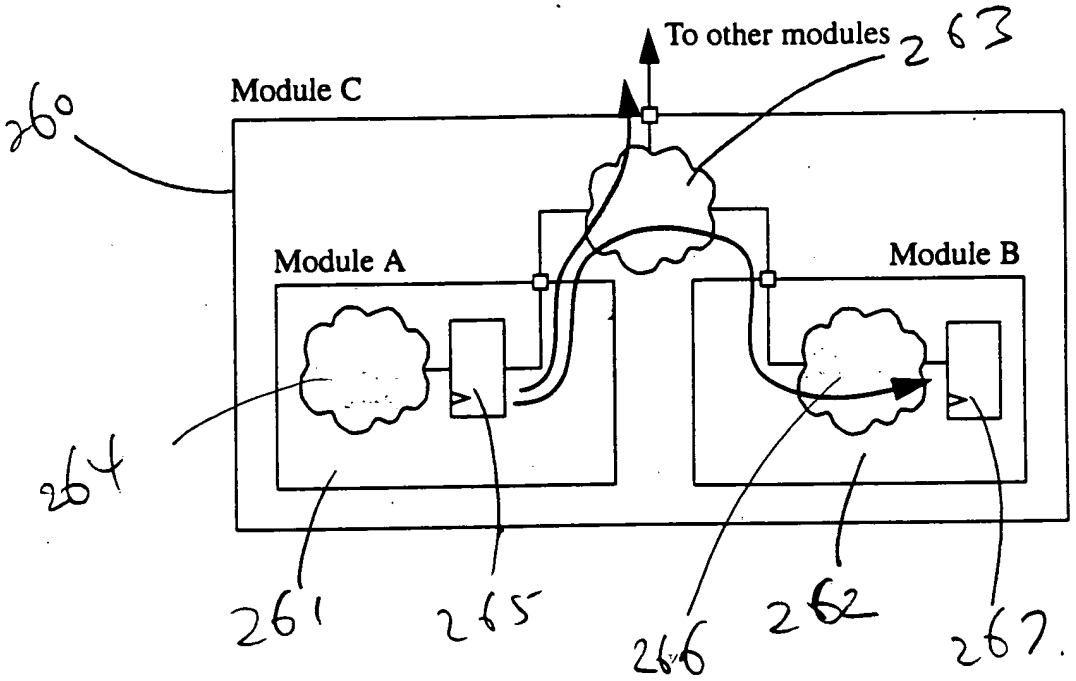


FIGURE 9

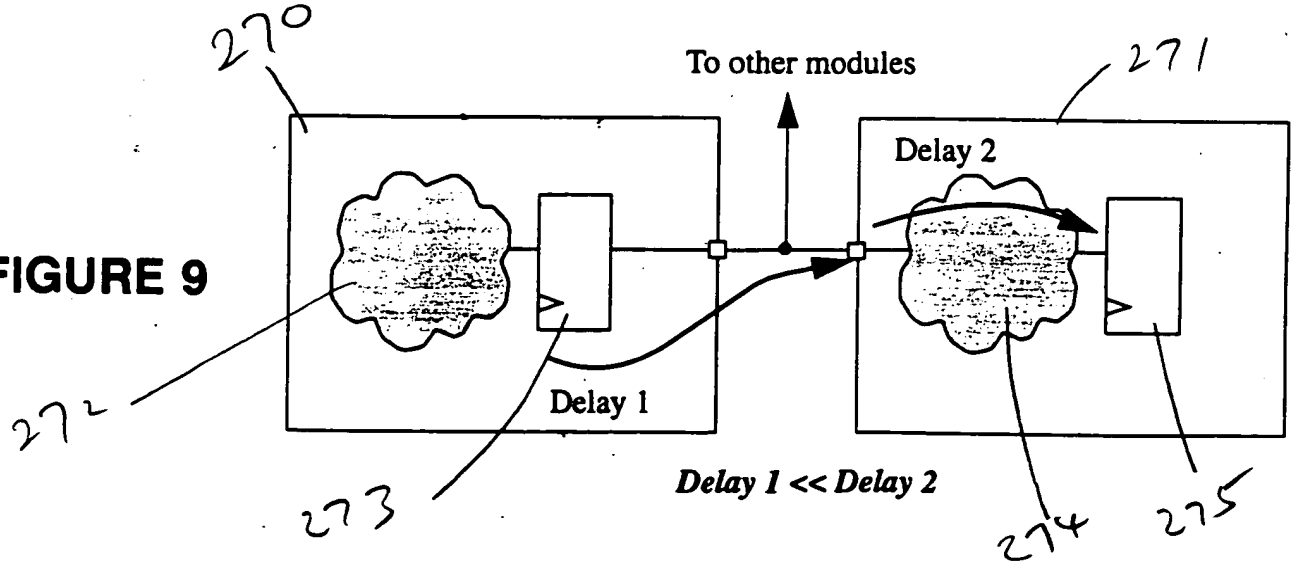


FIGURE 10A

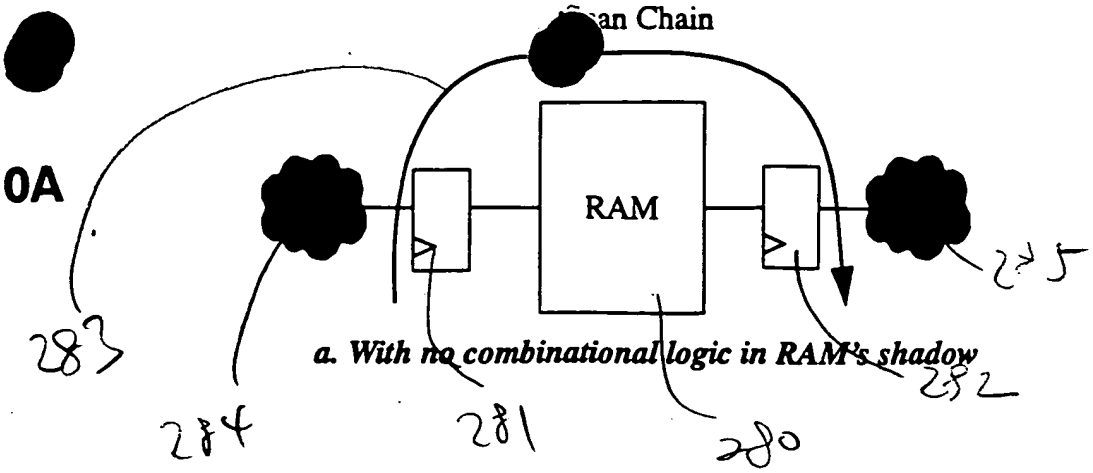


FIGURE 10B

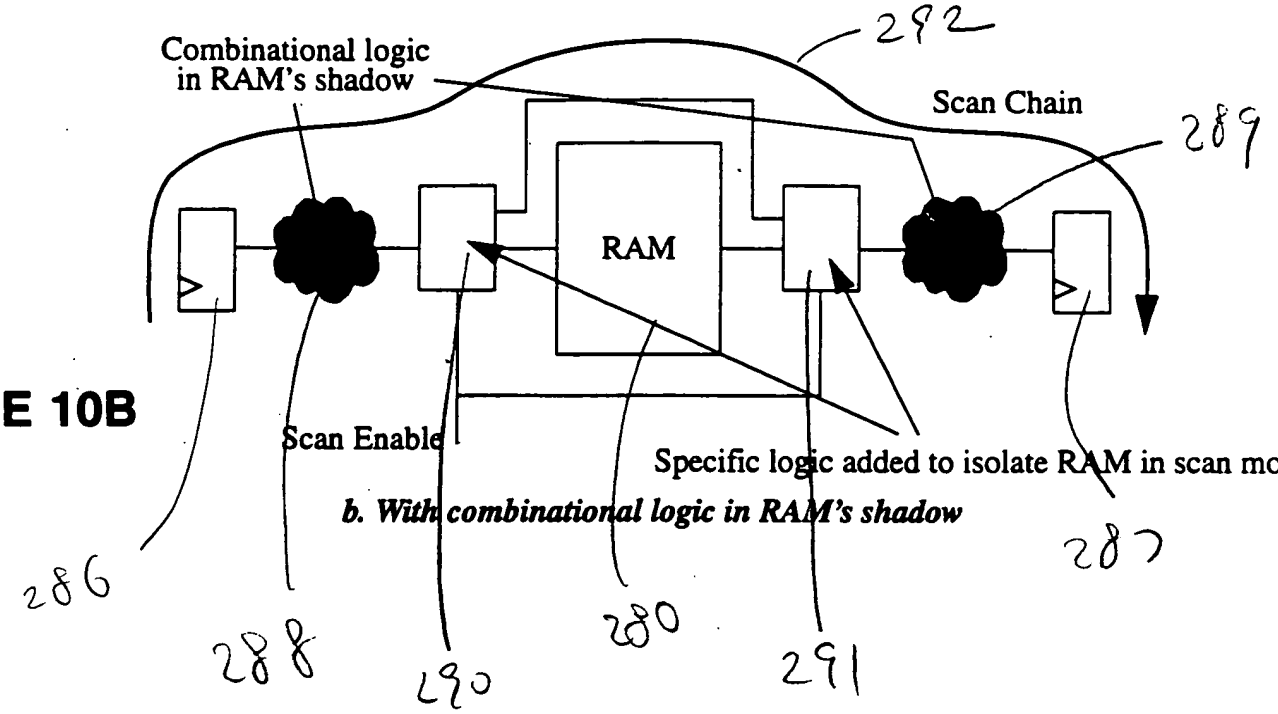


FIGURE 11

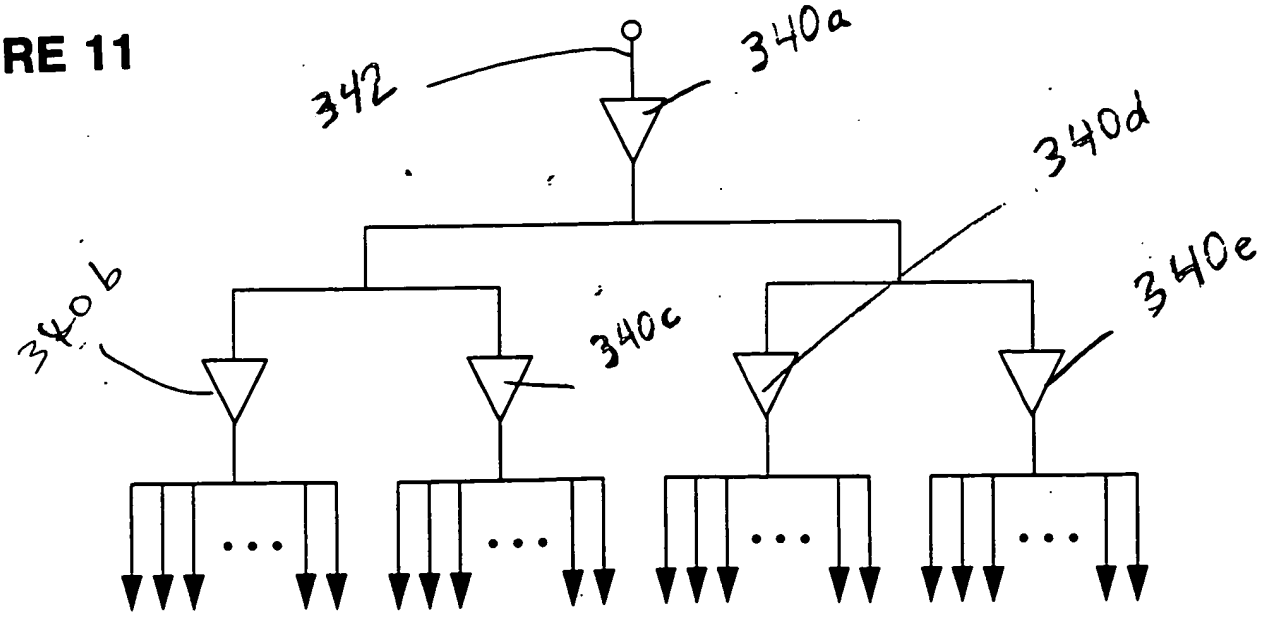


FIGURE 12

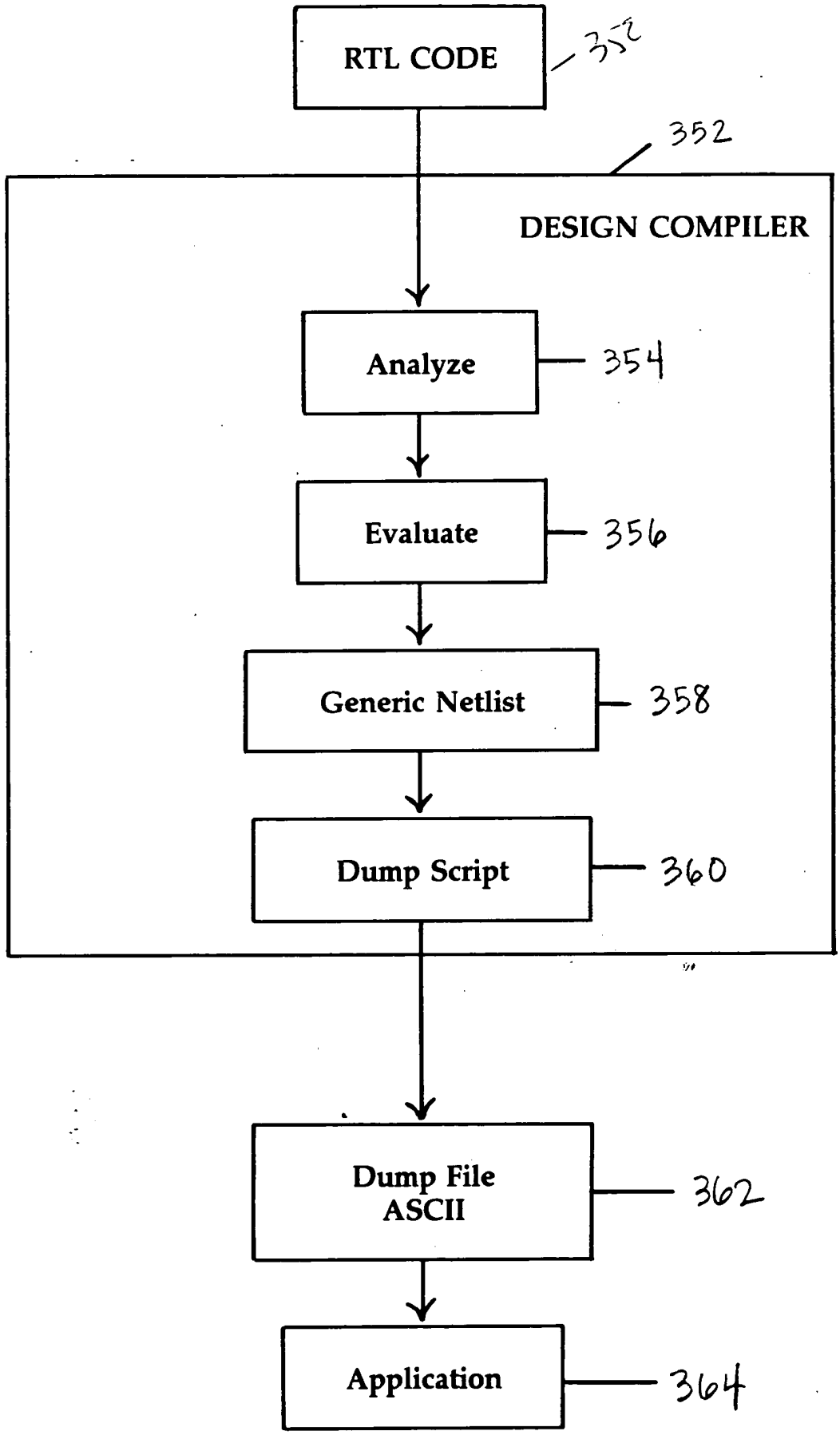


FIGURE 13

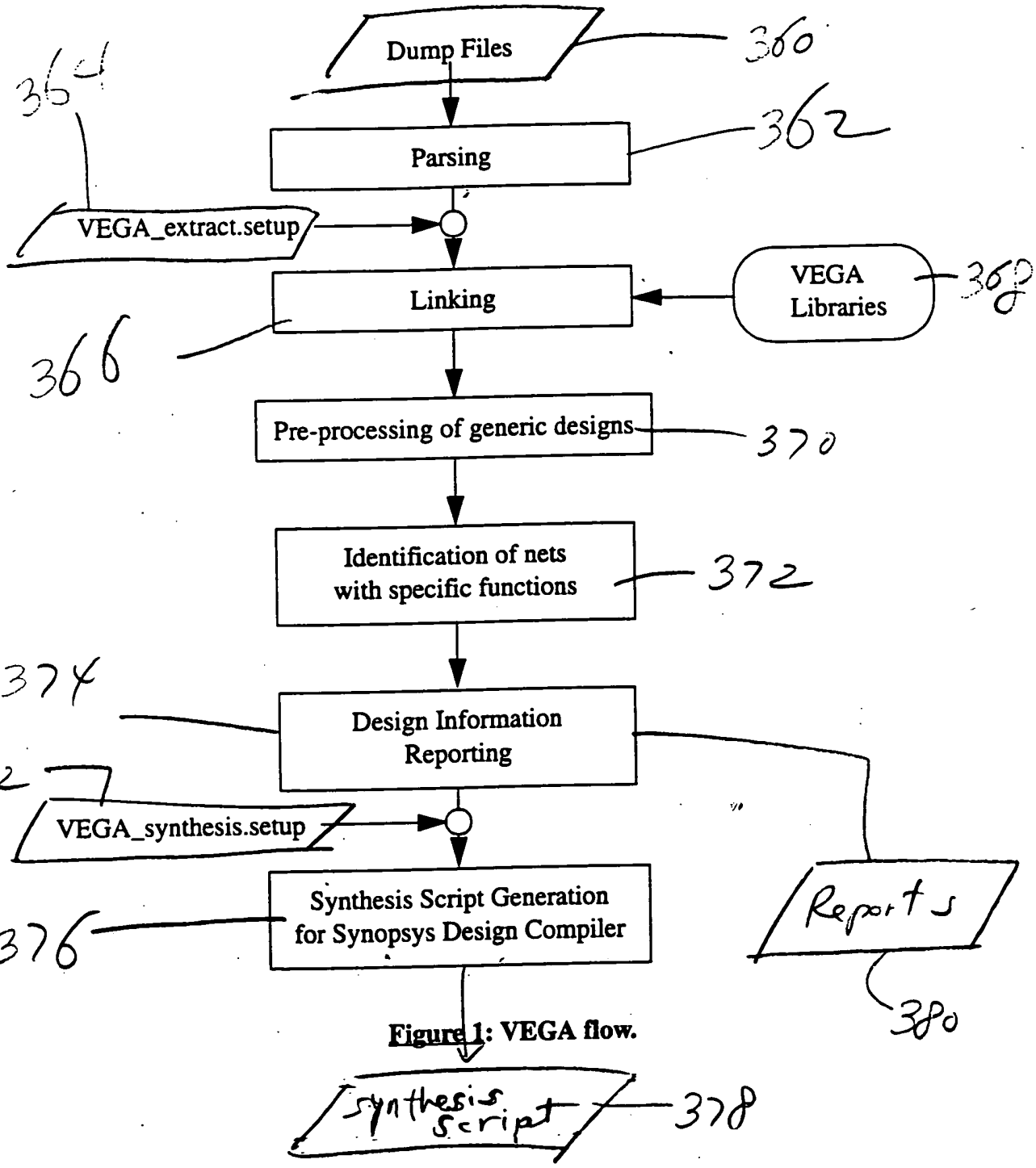
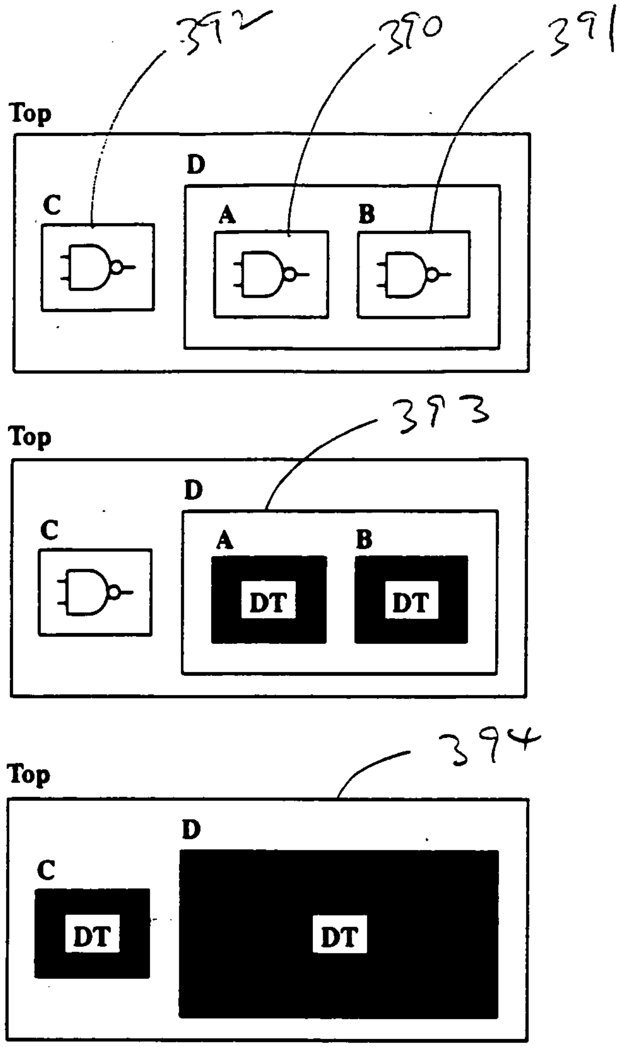


Figure 1: VEGA flow.



Step #1: Leaf modules A, B, and C are synthesized.

Step #2: Module D is synthesized with modules A and B made non-modifiable (dont-touch attributes).

Step #3: Module TOP is synthesized with modules C and D made non-modifiable.

FIGURE 14 : Bottom-up synthesis.

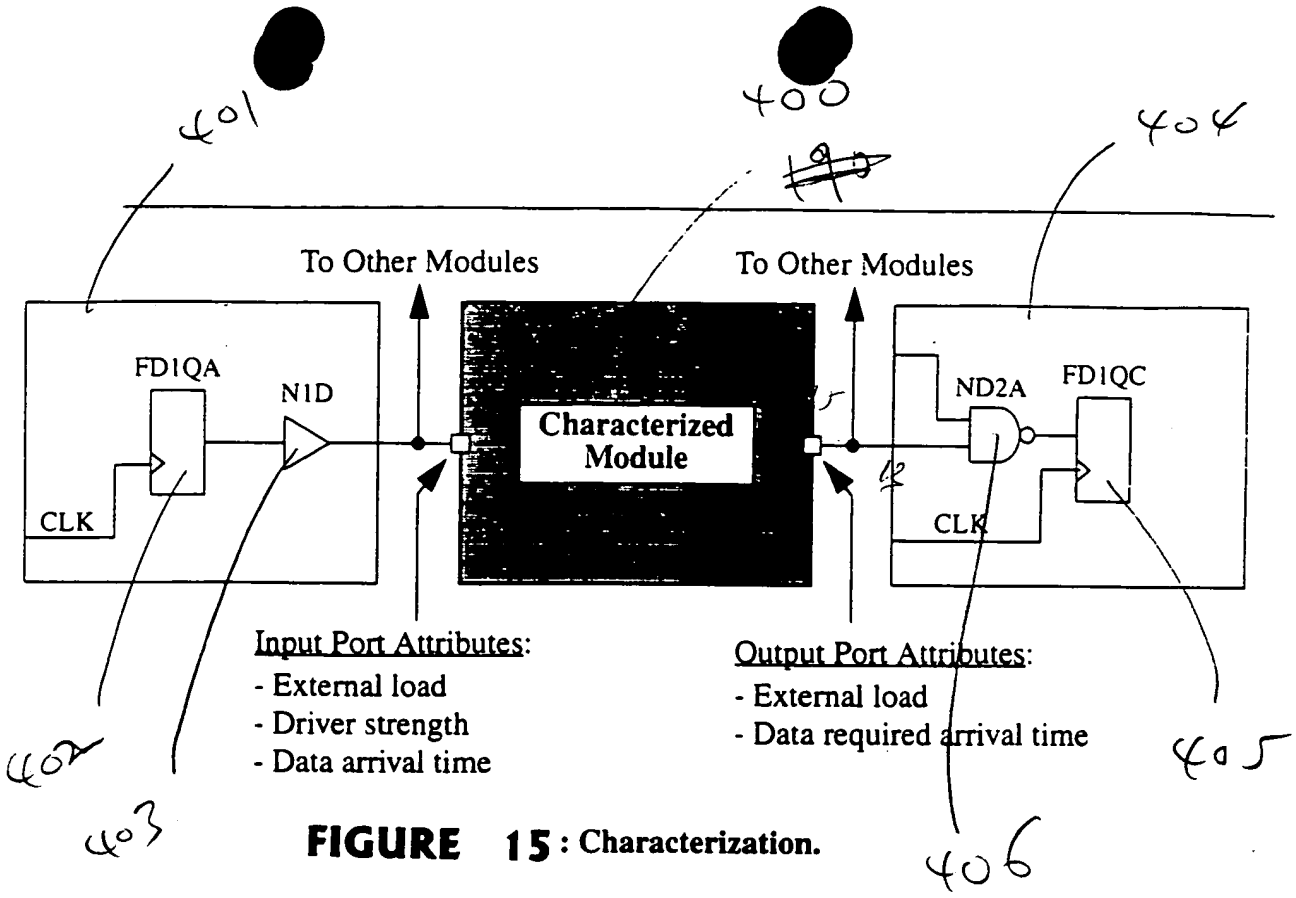


FIGURE 15 : Characterization.

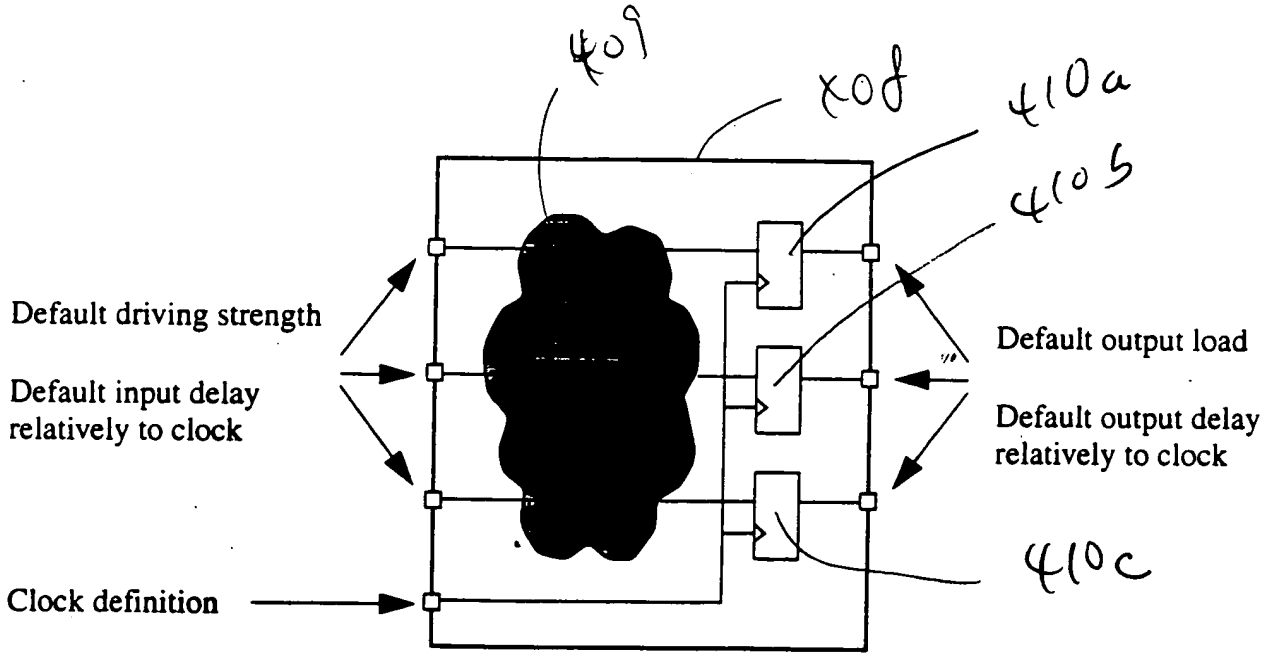


FIGURE 17 : Default constraints used for initial mapping.

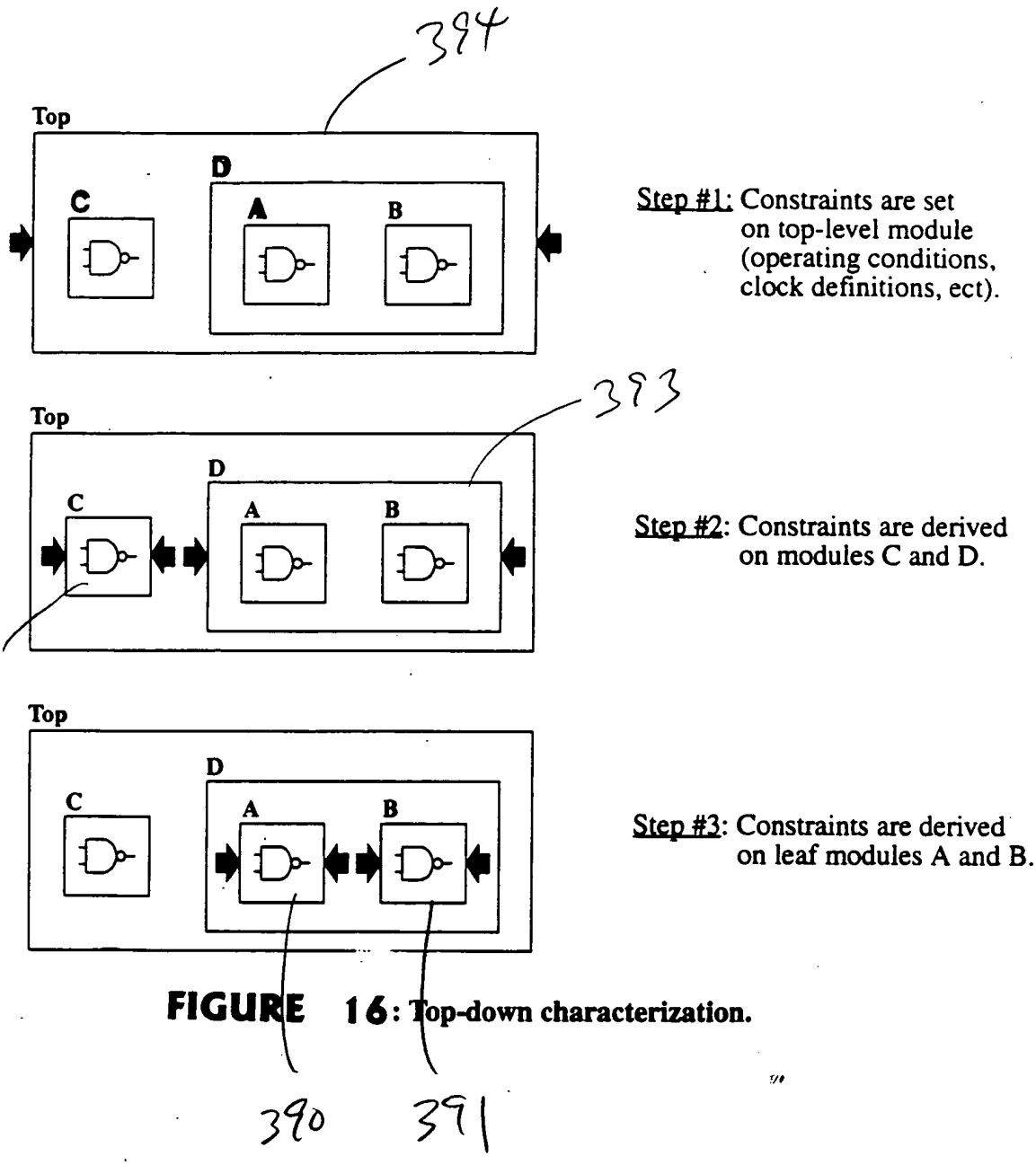


FIGURE 16: Top-down characterization.

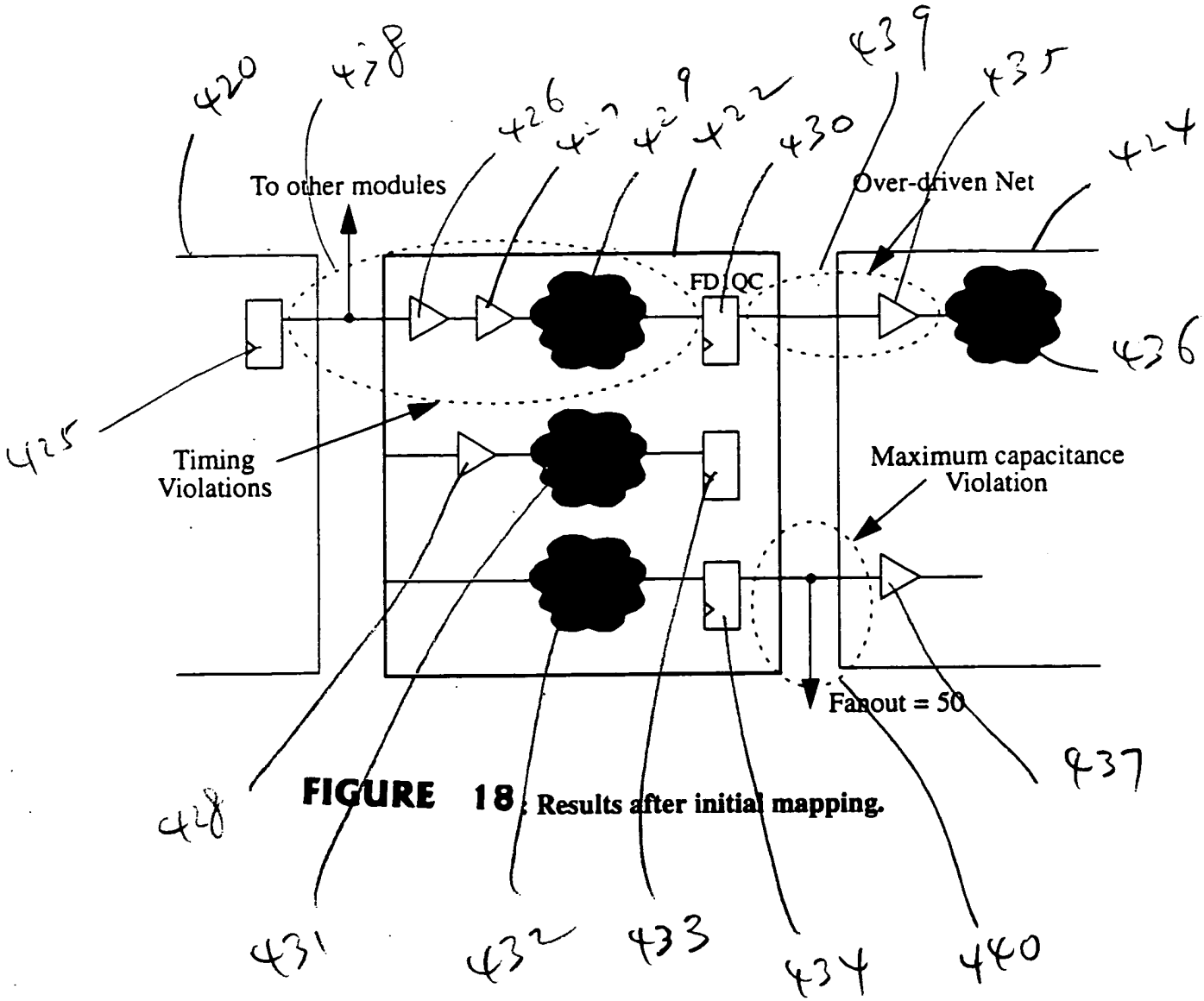


FIGURE 18: Results after initial mapping.

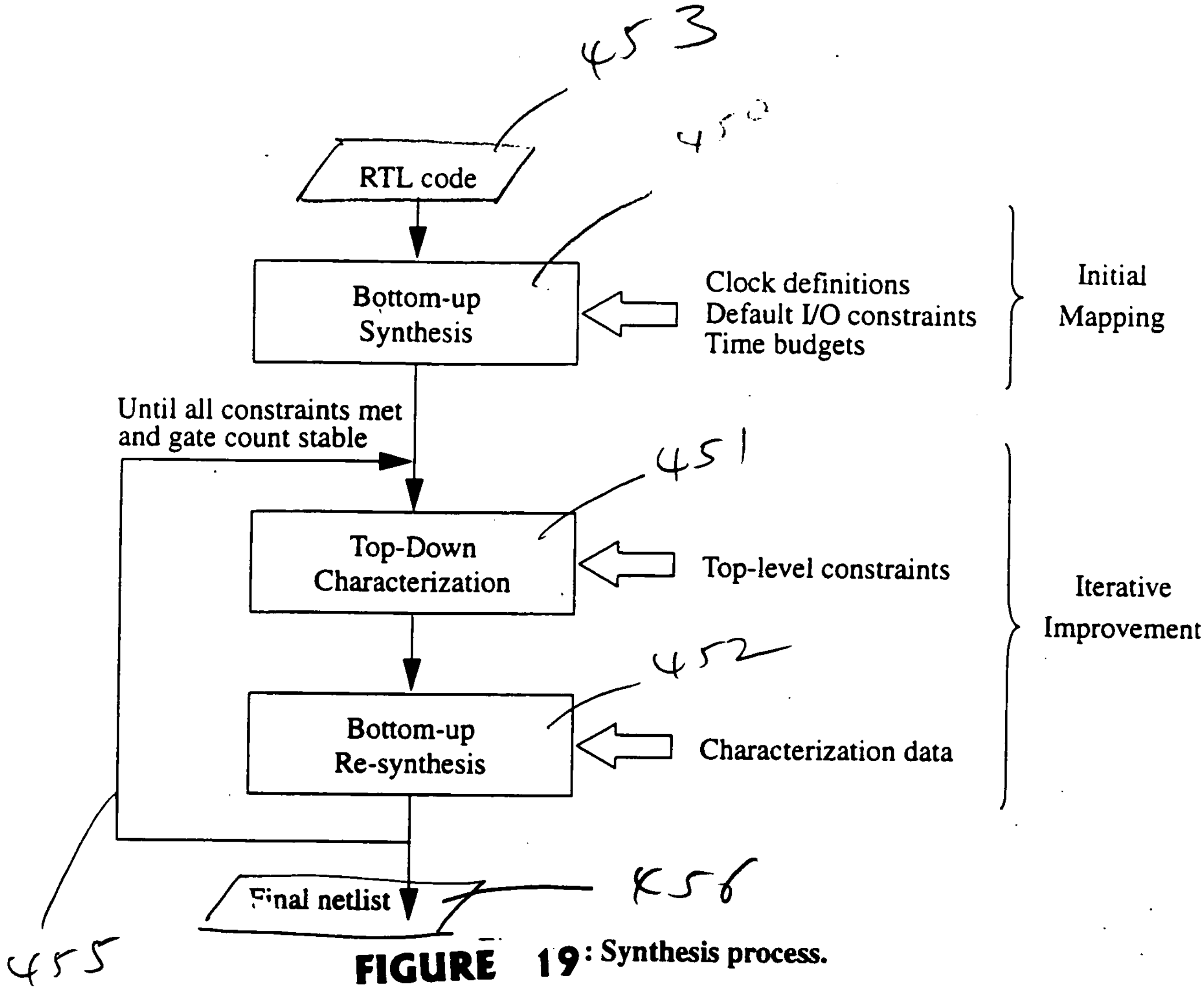


FIGURE 19: Synthesis process.

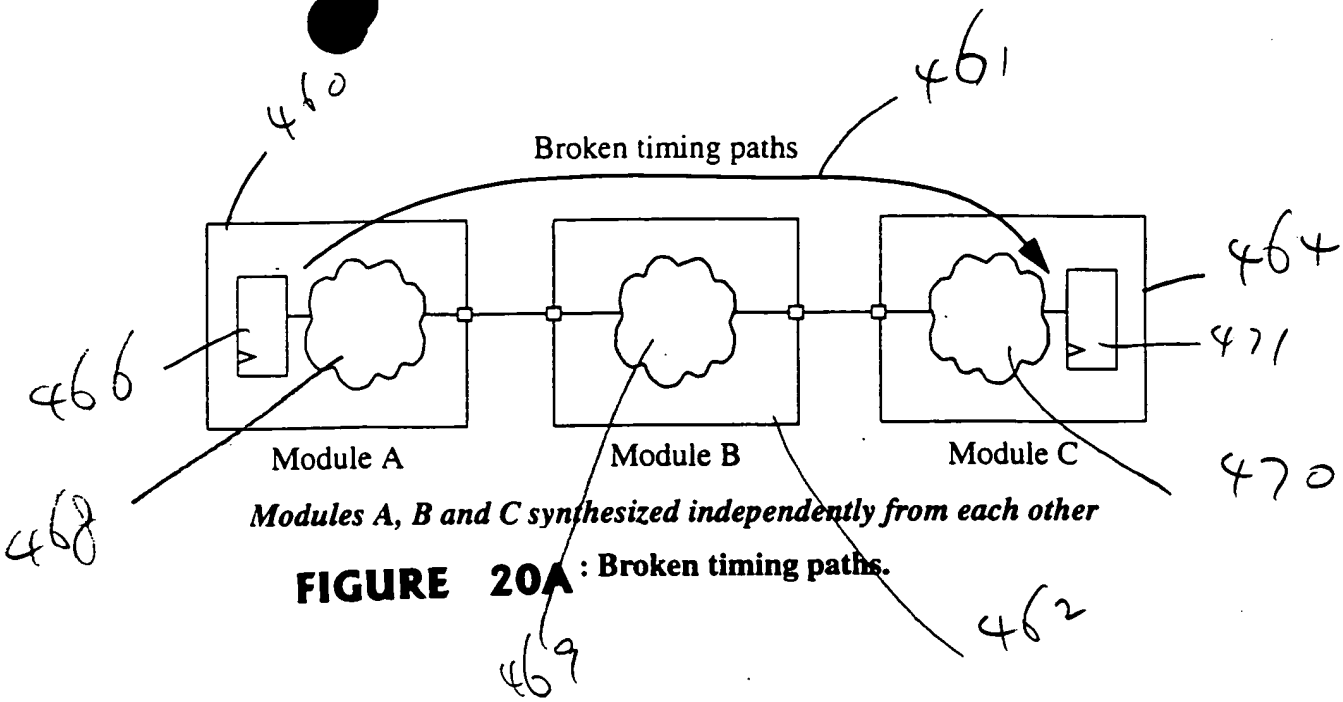


FIGURE 20A: Broken timing paths.

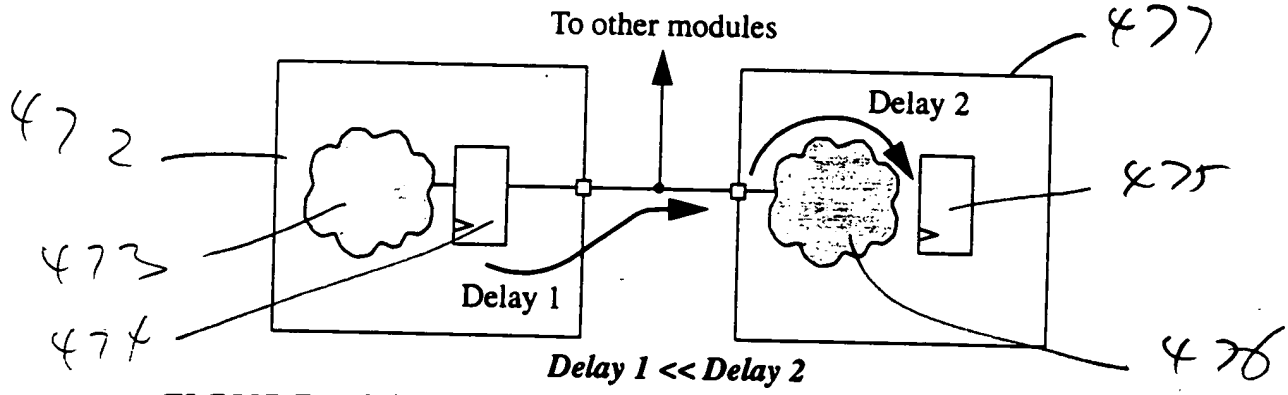


FIGURE 20B: In the absence of broken timing paths.

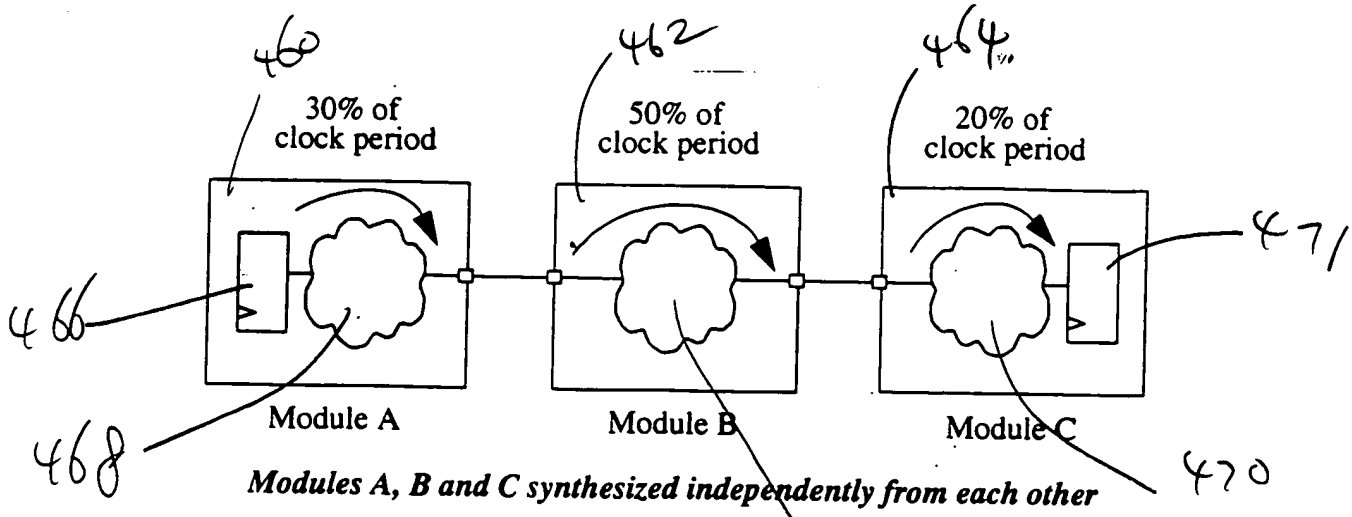


FIGURE 20C: Time budgets.

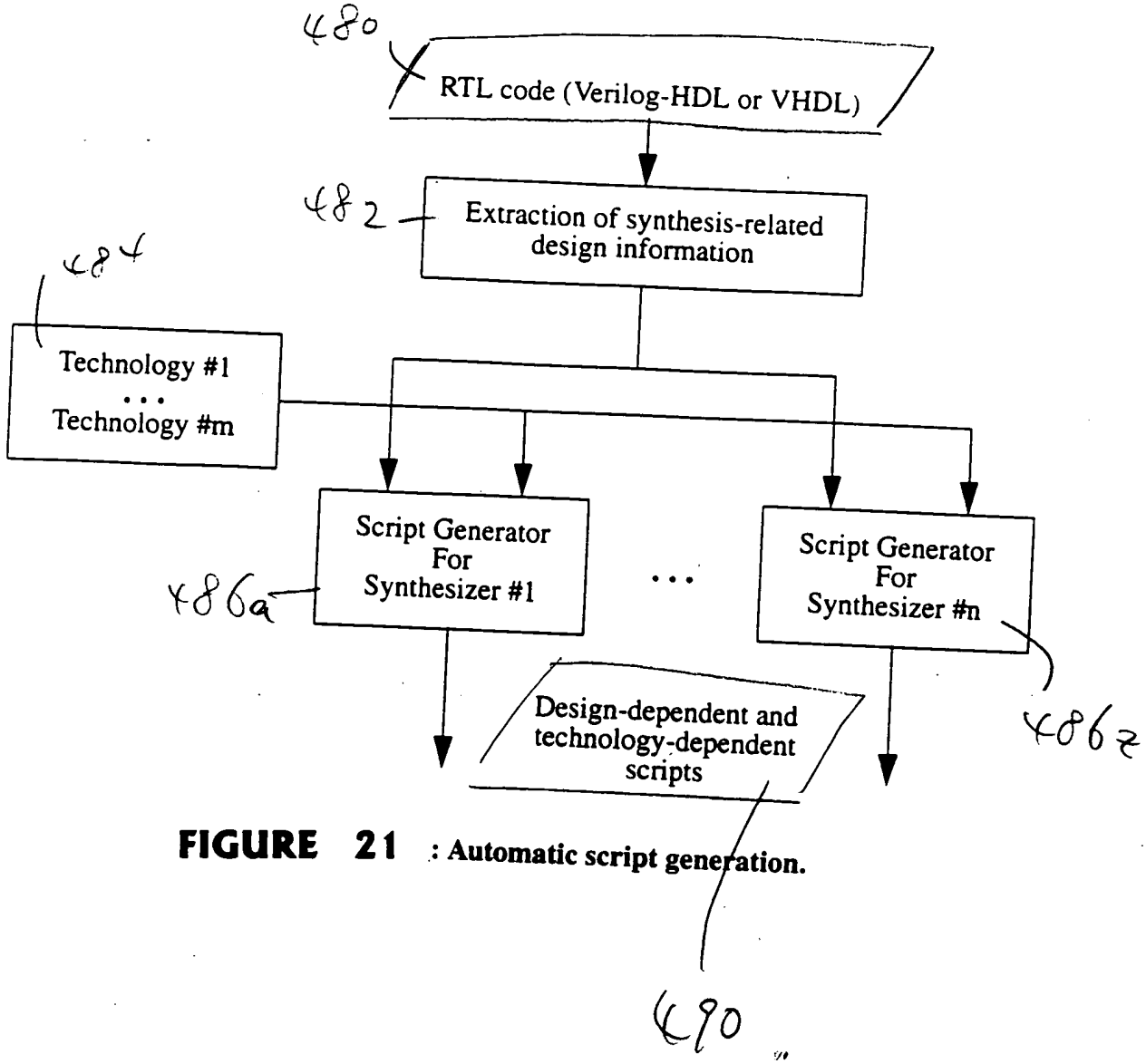


FIGURE 21 : **Automatic script generation.**

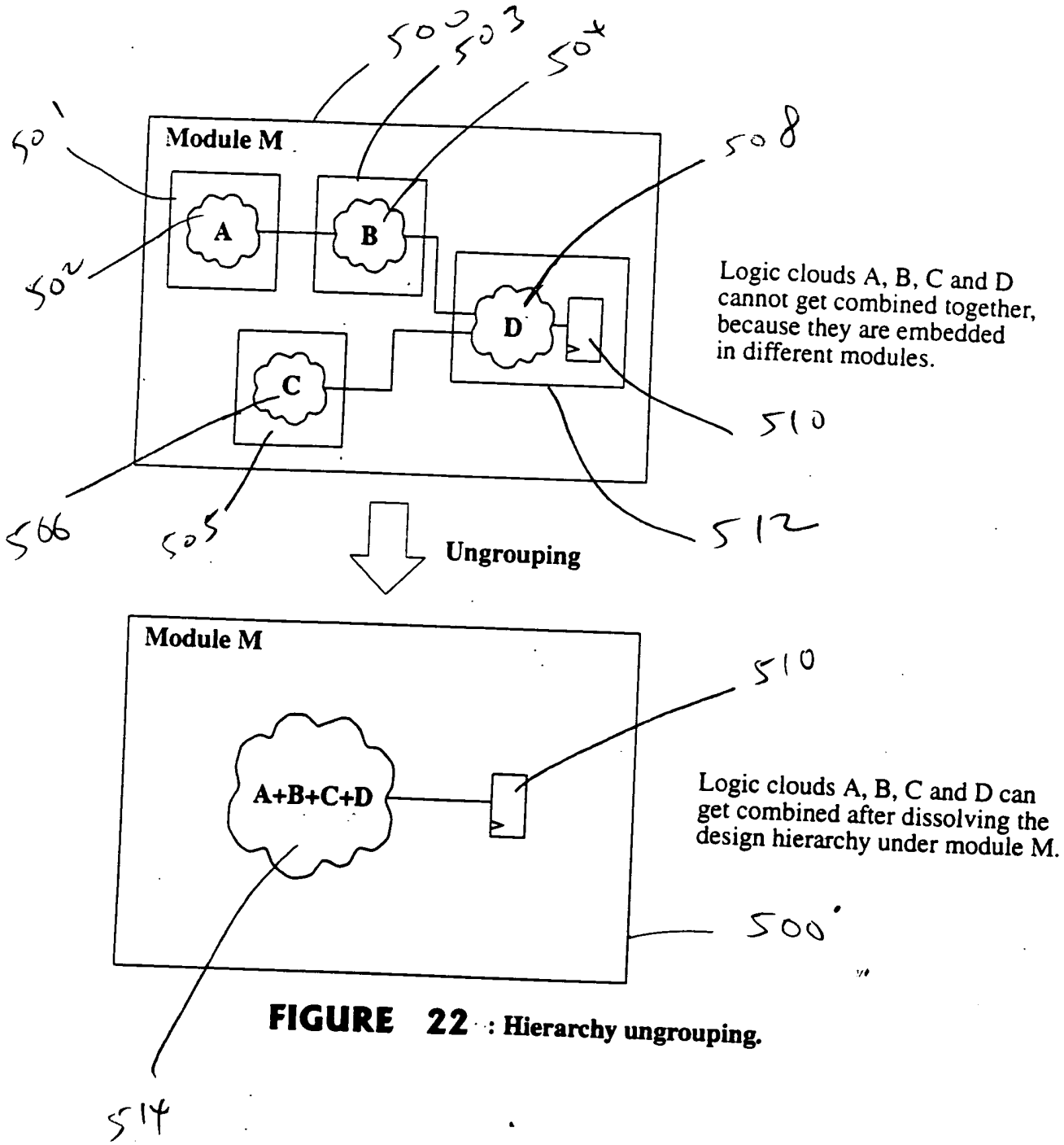
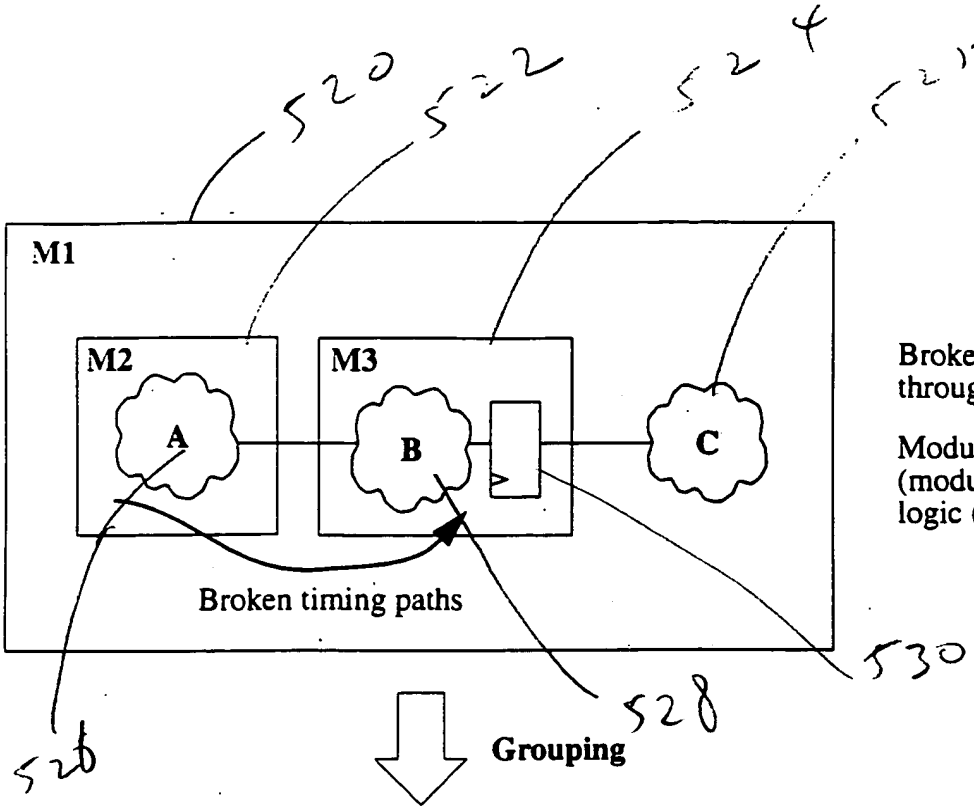
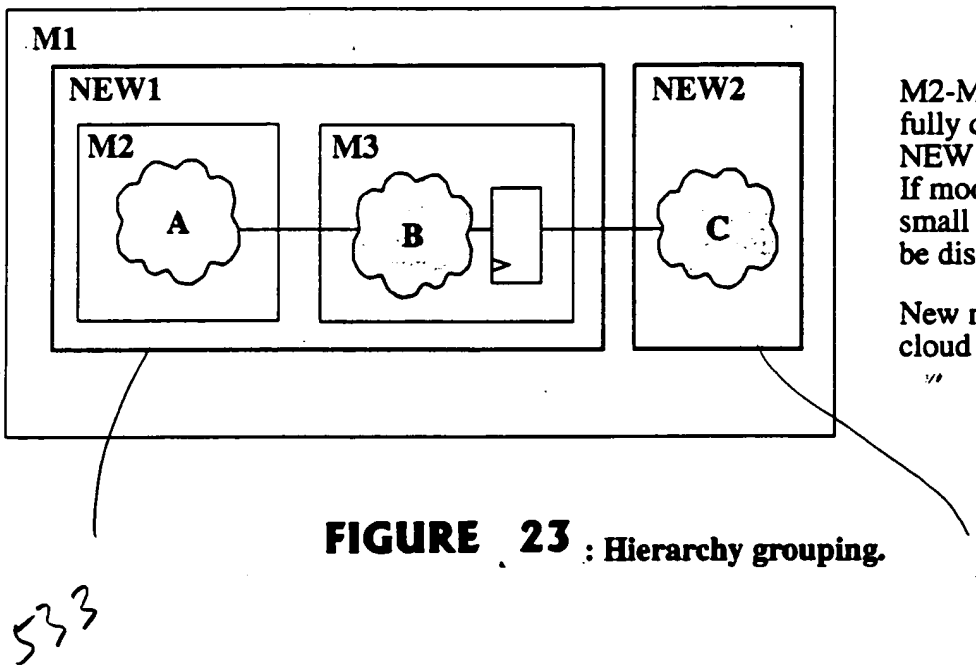


FIGURE 22 : Hierarchy ungrouping.



Broken timing paths run through modules M2 and M3.

Module M1 mixes hierarchy (modules M2 and M3) with logic (cloud C).



M2-M3 timing paths are now fully contained in new module NEW1.
If modules M2 and M3 are small enough, the hierarchy can be dissolved below NEW1.

New module NEW2 encapsulates cloud of logic C.

FIGURE 23 : Hierarchy grouping.

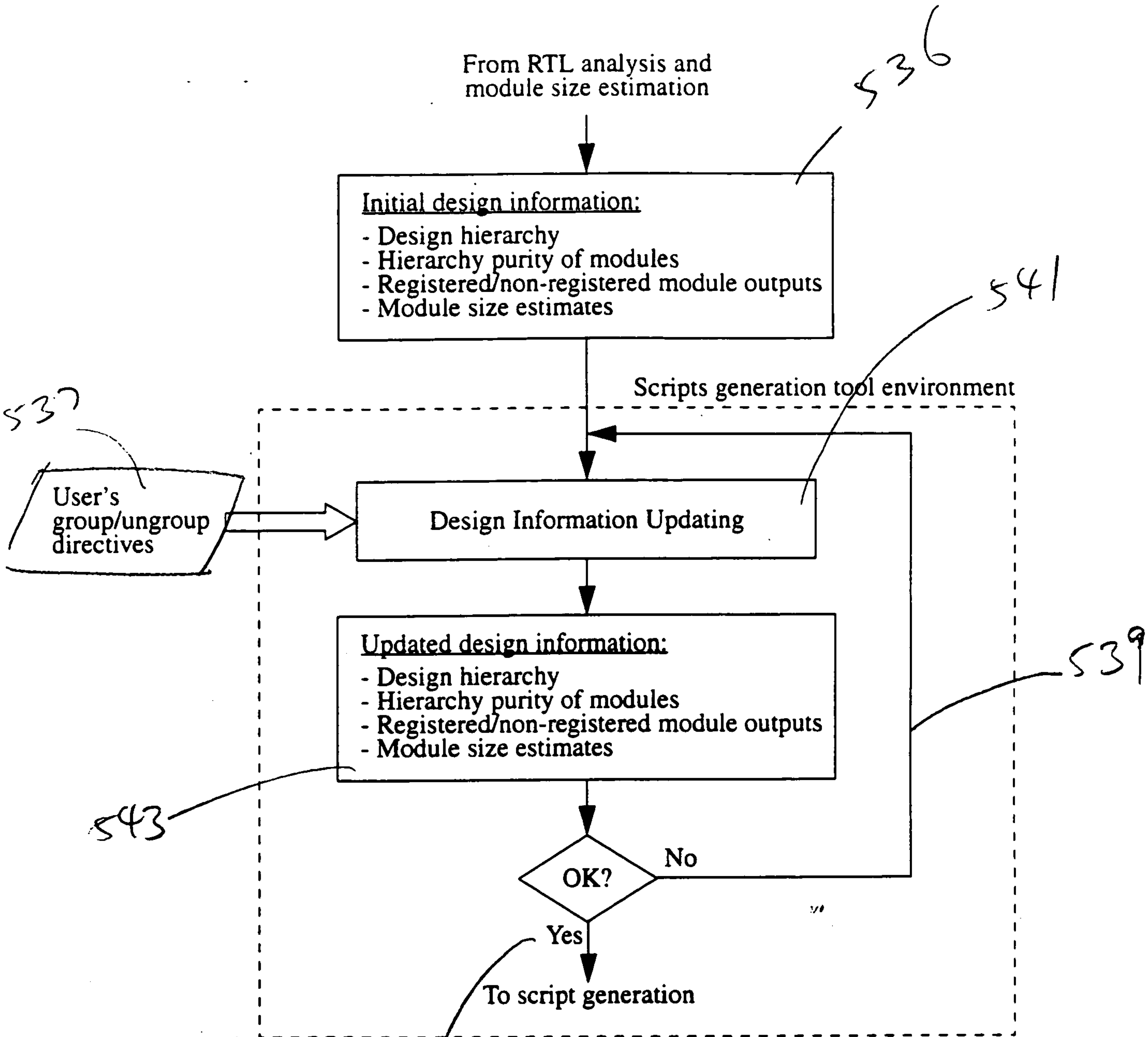


FIGURE 24 : **Support for design hierarchy re-arrangement.**

disclosure 2 fig. 14

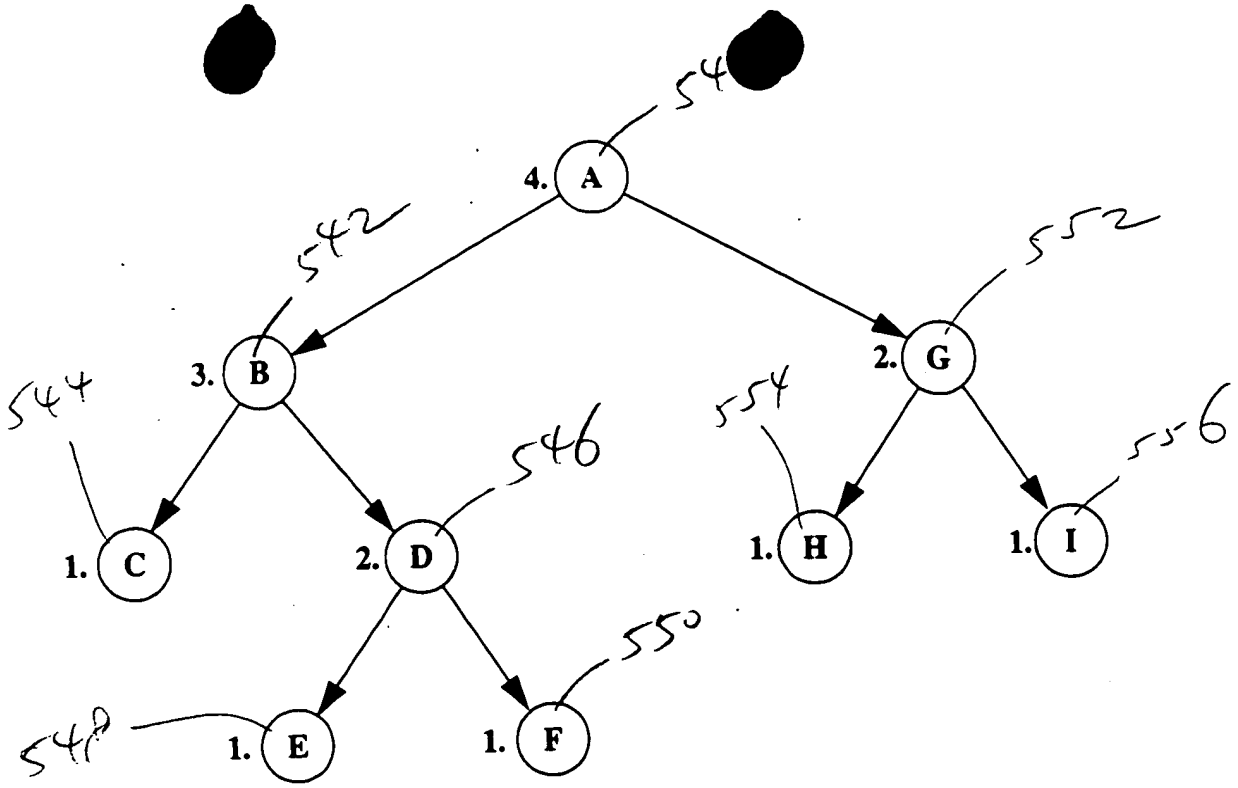


FIGURE 25 : Module processing order for parallel bottom-up synthesis.

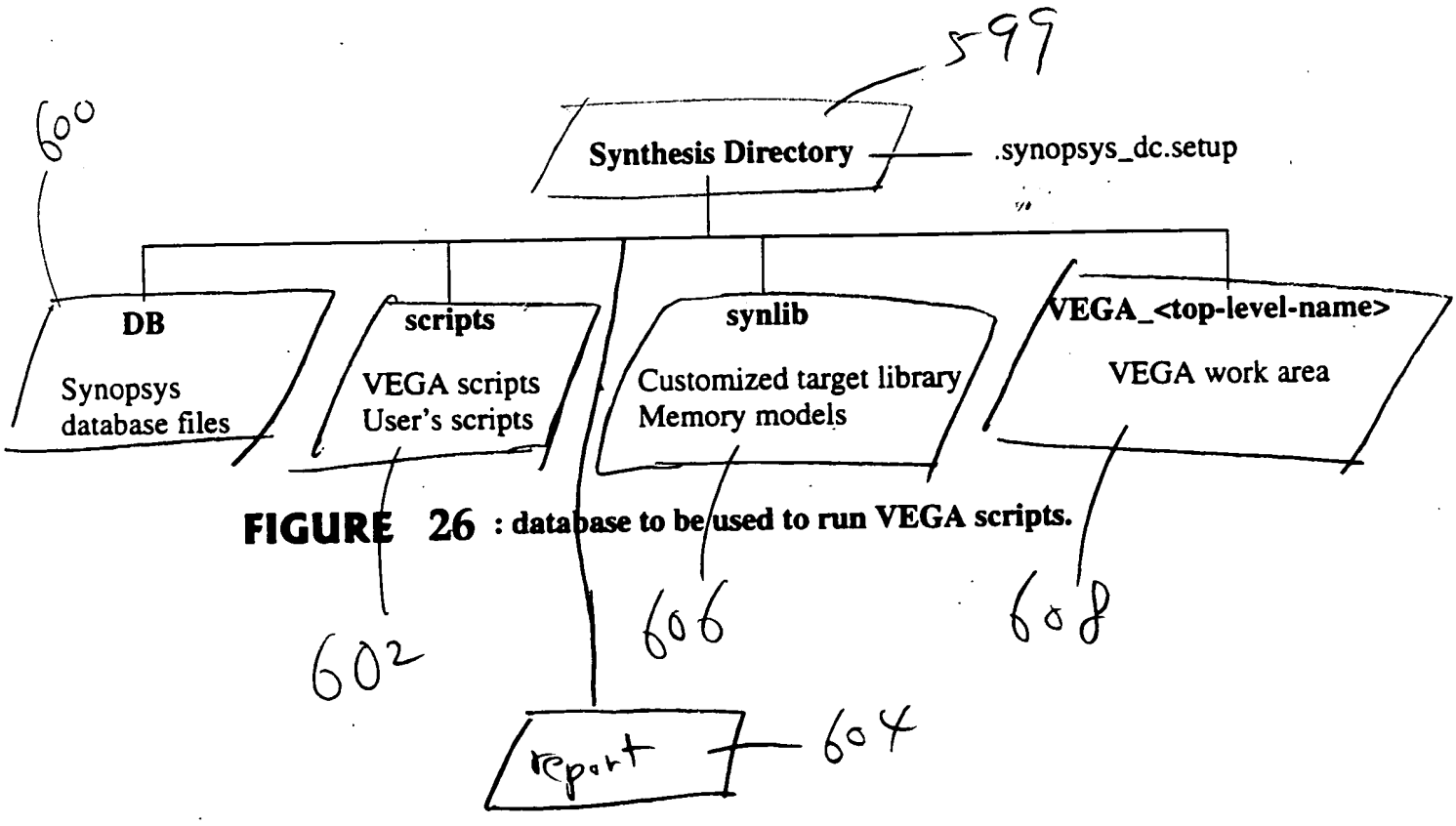


FIGURE 26 : database to be used to run VEGA scripts.

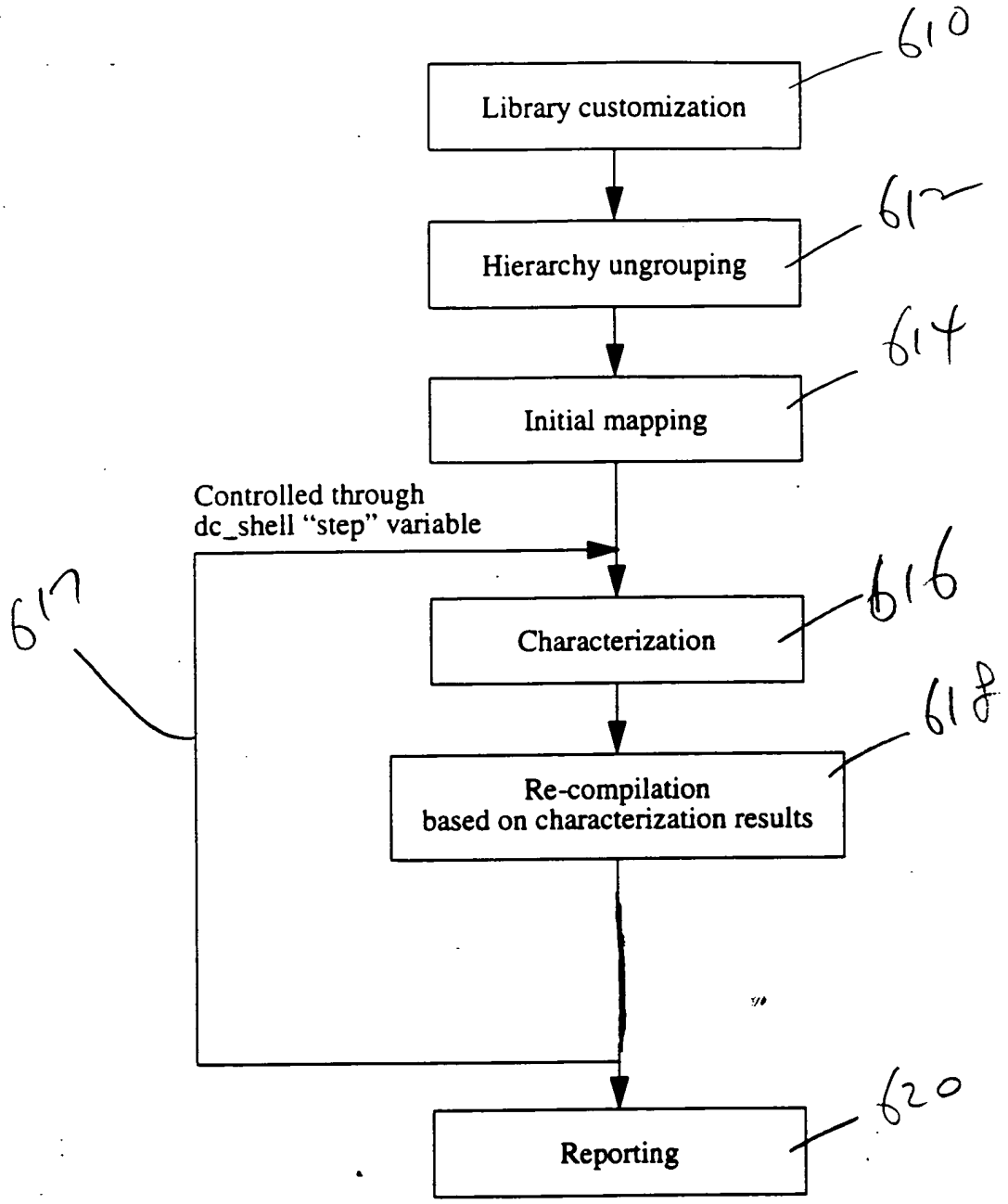


FIGURE 27 : Script flow implemented by VEGA.

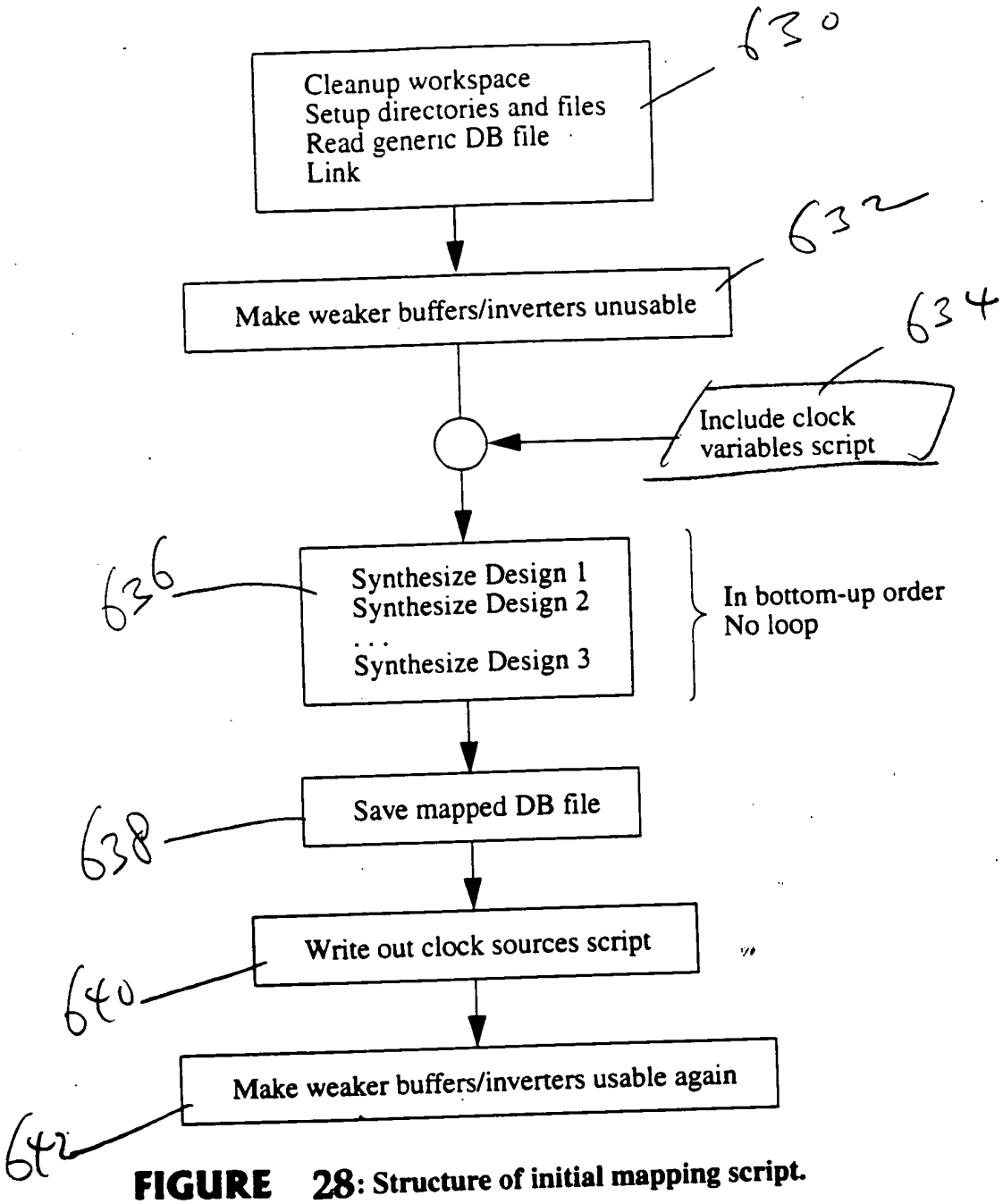


FIGURE 28: Structure of initial mapping script.

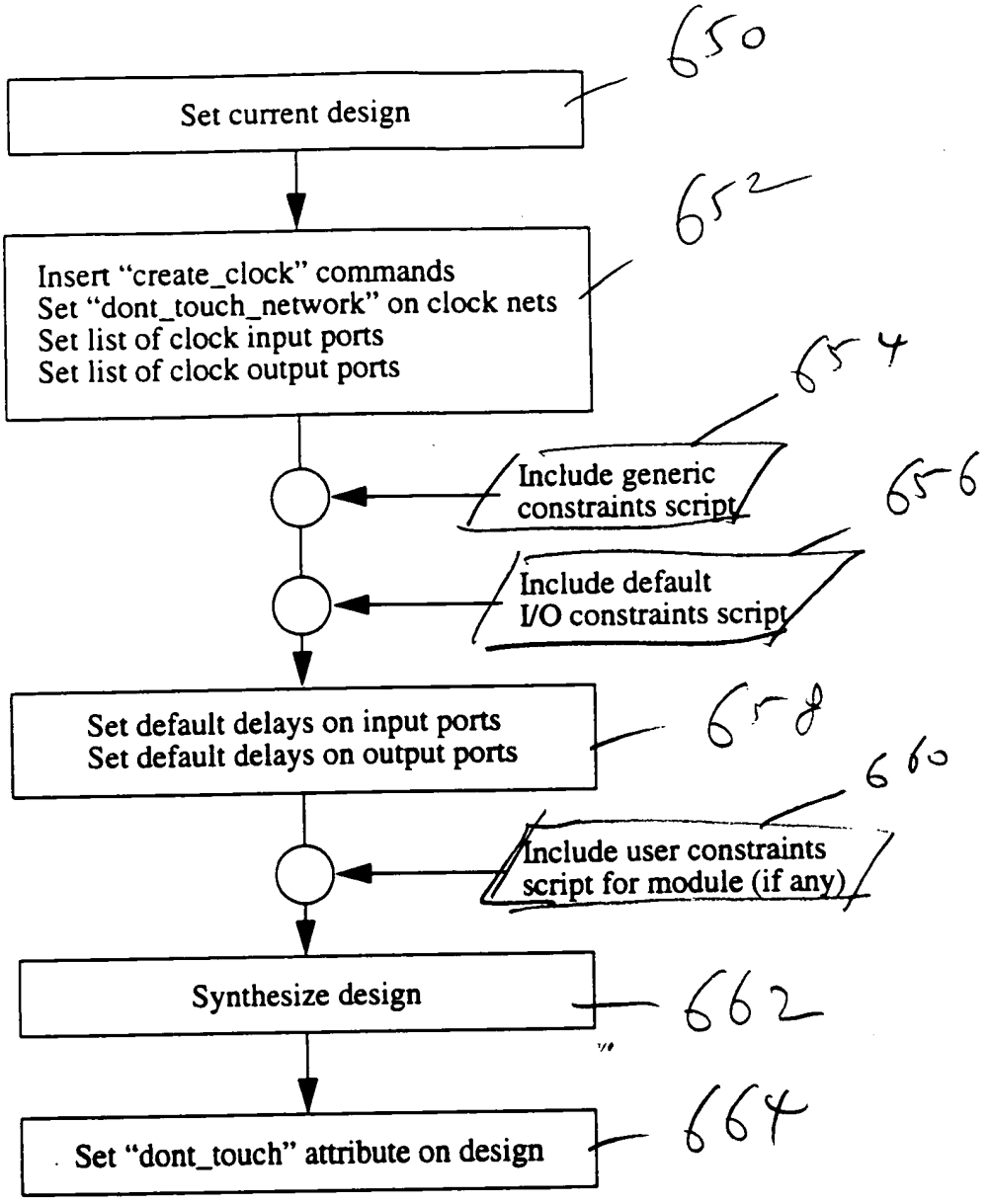


FIGURE 29 : Operations performed on each module by initial mapping.

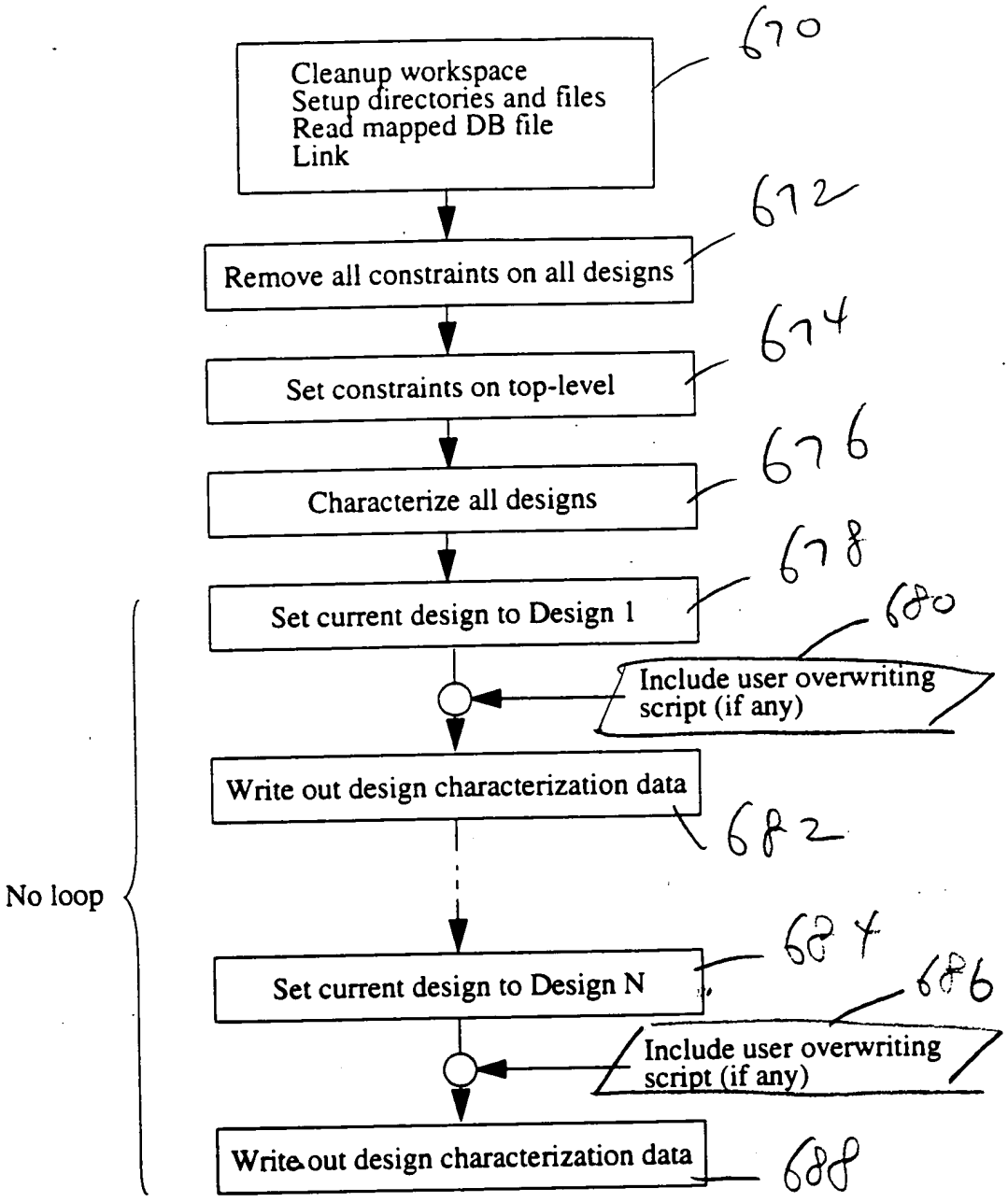


FIGURE 30 : Structure of characterization script.

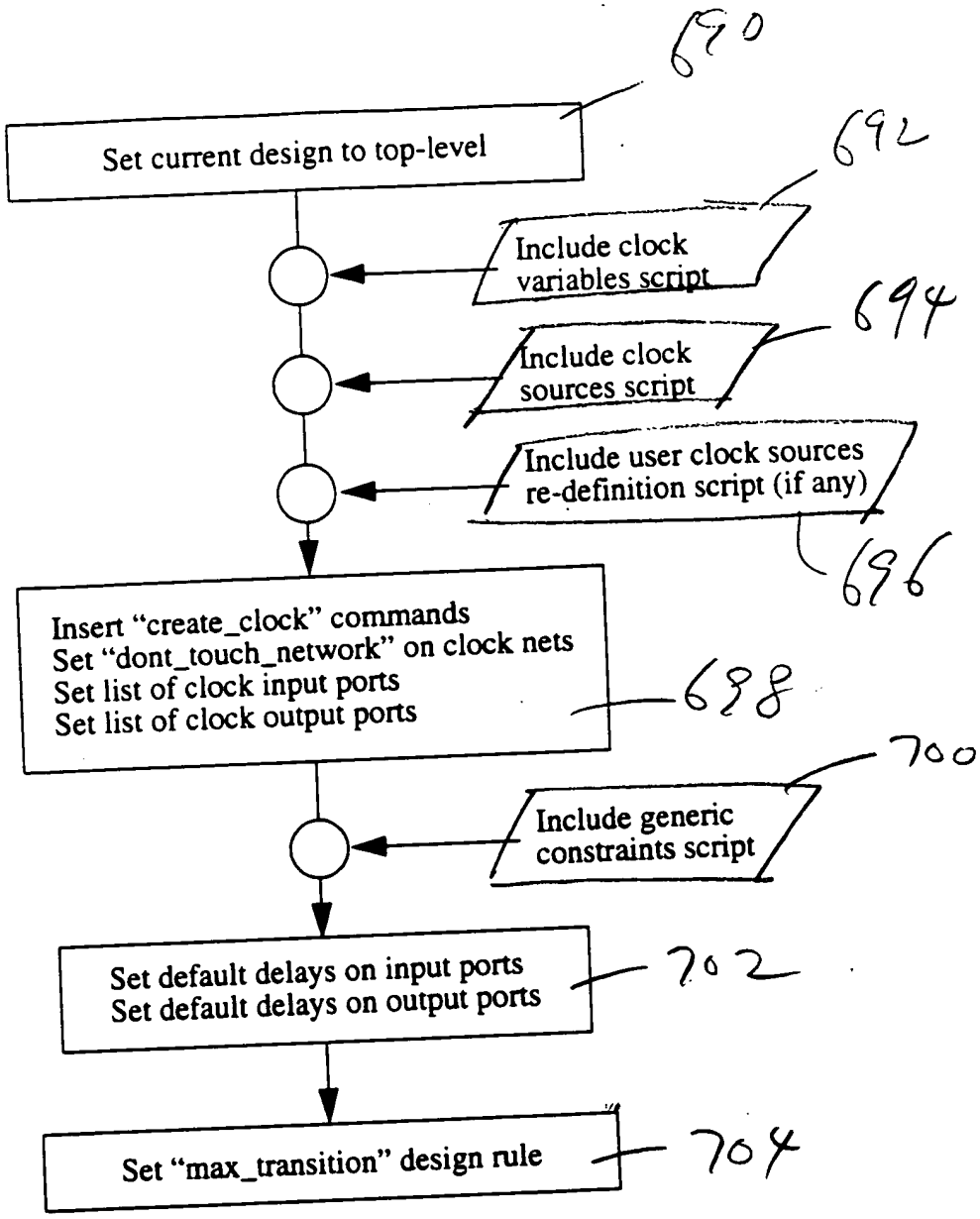


FIGURE 31 : Structure of constraints setting on top-level.

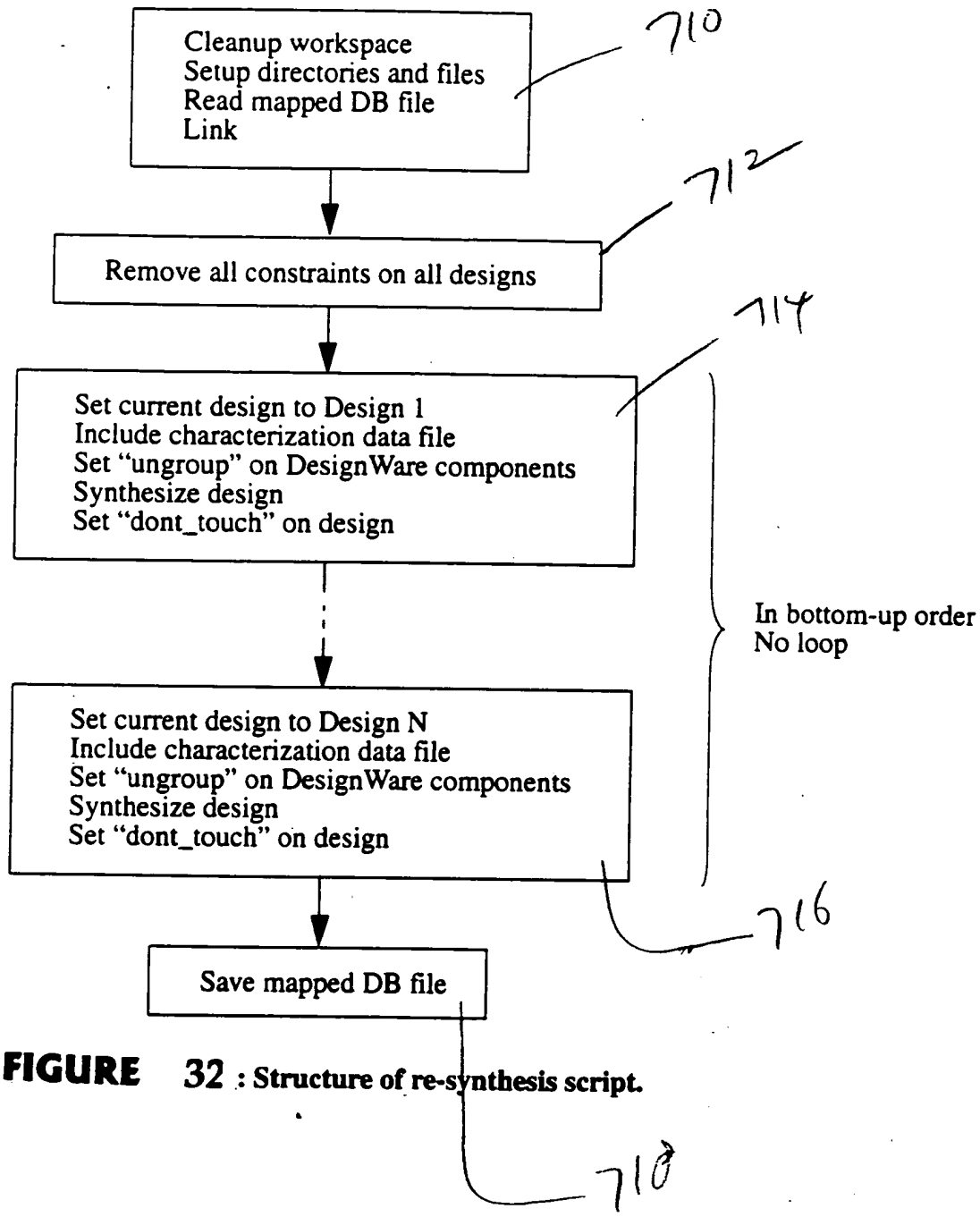


FIGURE 32 : Structure of re-synthesis script.

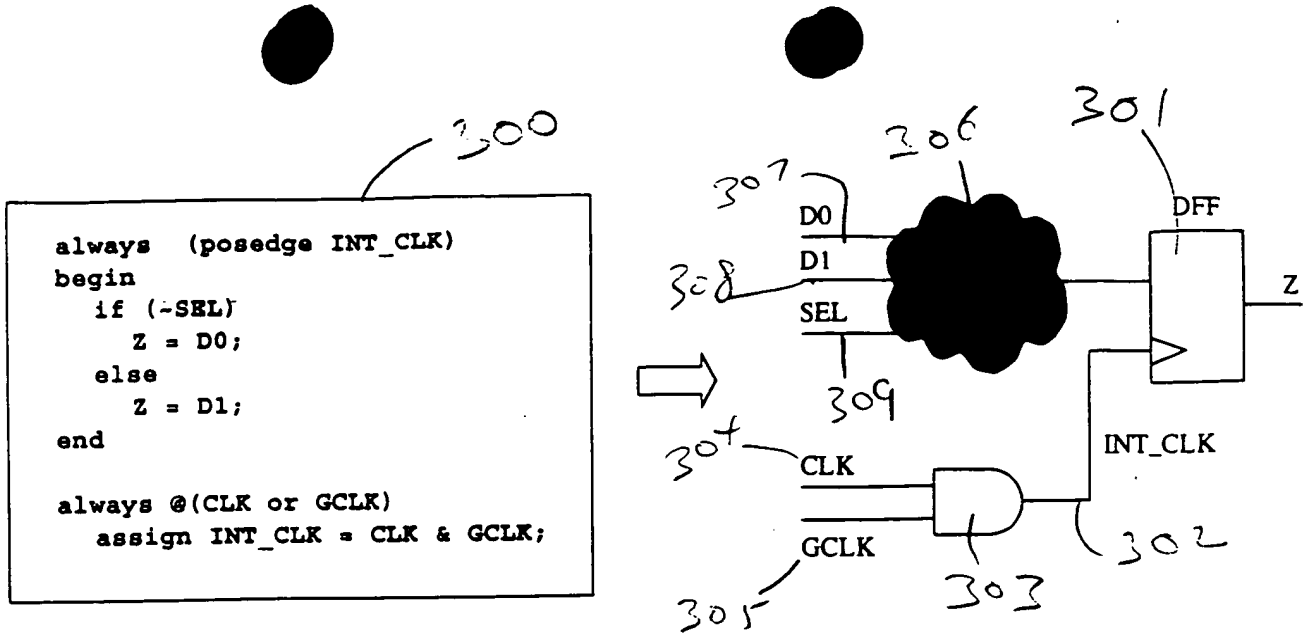


FIGURE 33 : Example of RTL code and equivalent Hardware view for RTL analysis.

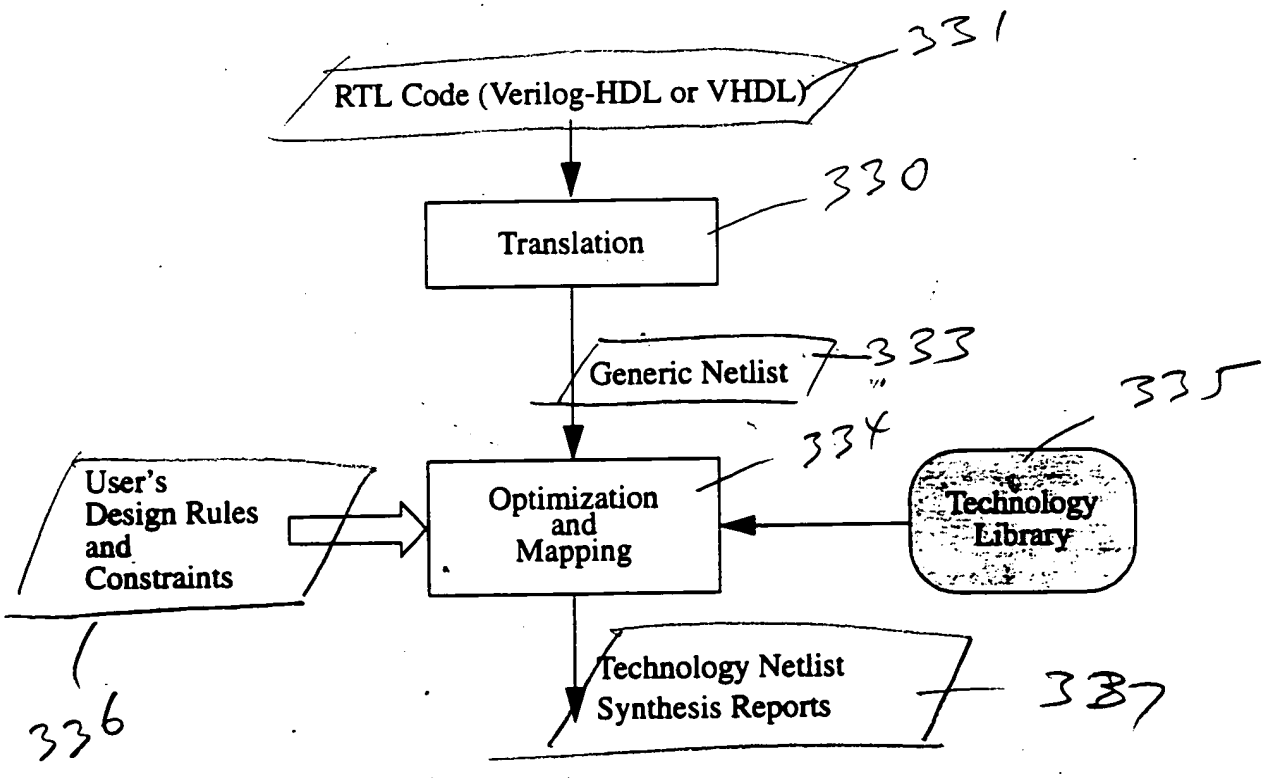


FIGURE 36 : Logic synthesis process

FIGURE 34A

310

```
process(RESET_N, CLK)
begin
   if (RESET_N = '0') THEN
      Q1 <= '0';
   elsif CLK'event AND (CLK = '1') THEN
      Q1 <= D1;
      Q2 <= D2;
   endif;
end process;
```

a. VHDL code for a 2-bit register with partial asynchronous reset

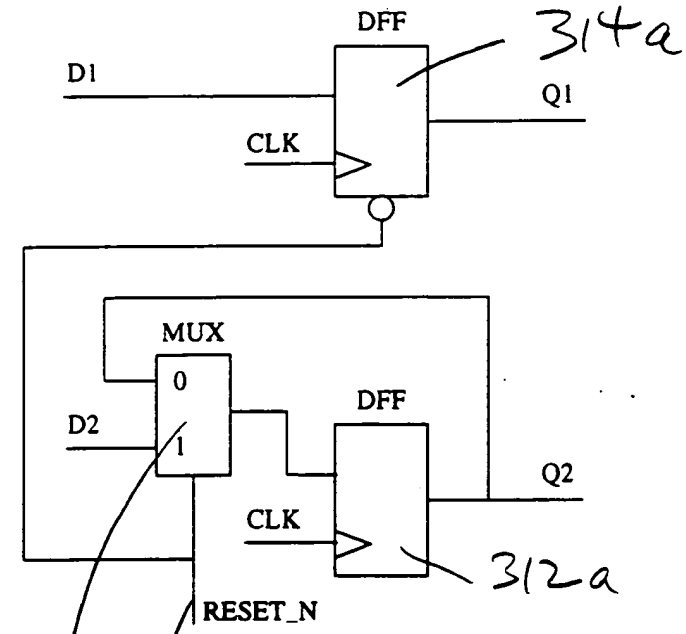


FIGURE 34B

b. Implementation created by Synopsys Design Compiler.

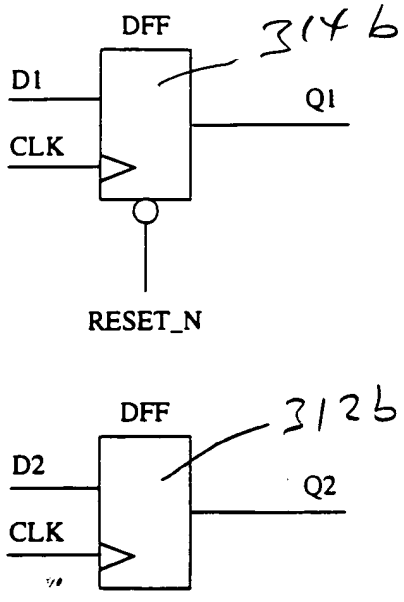


FIGURE 34C

c. Implementation created by AMBIT BuildGates

: Implementation of partial asynchronous reset.

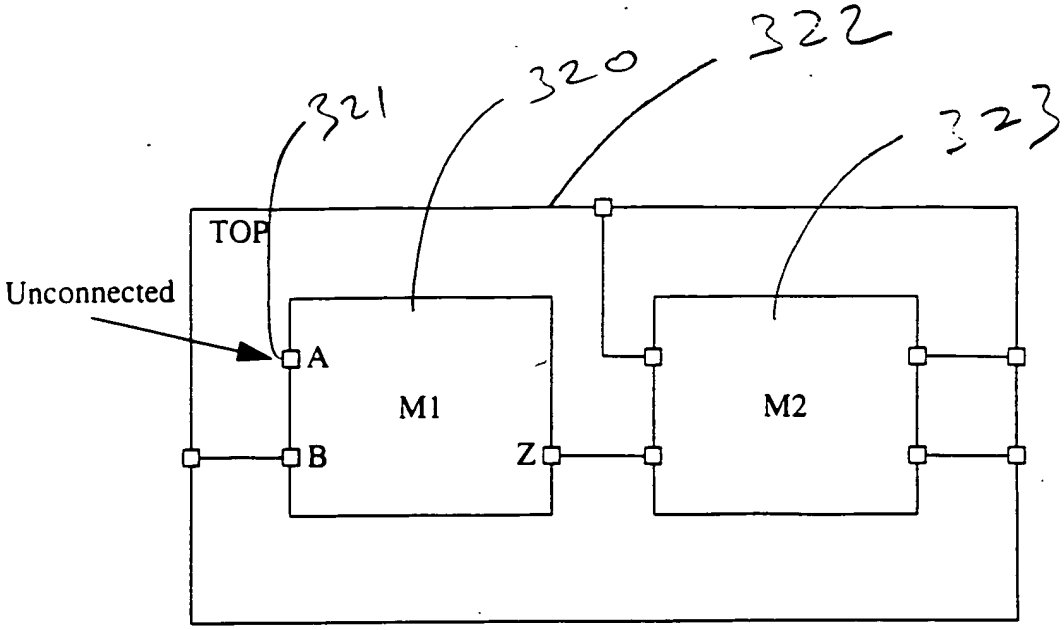


FIGURE 35 A . RTL code

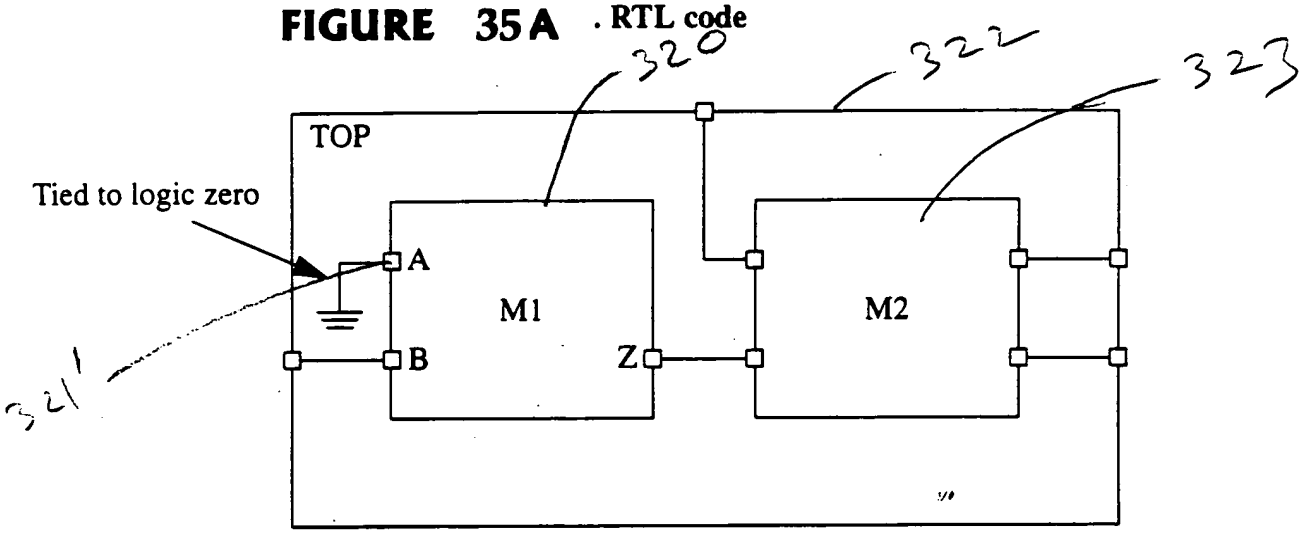


FIGURE 35 B Synopsys Design Compiler view of the RTL code

; **Handling of unconnected module input pins by Synopsys Design Compiler.**

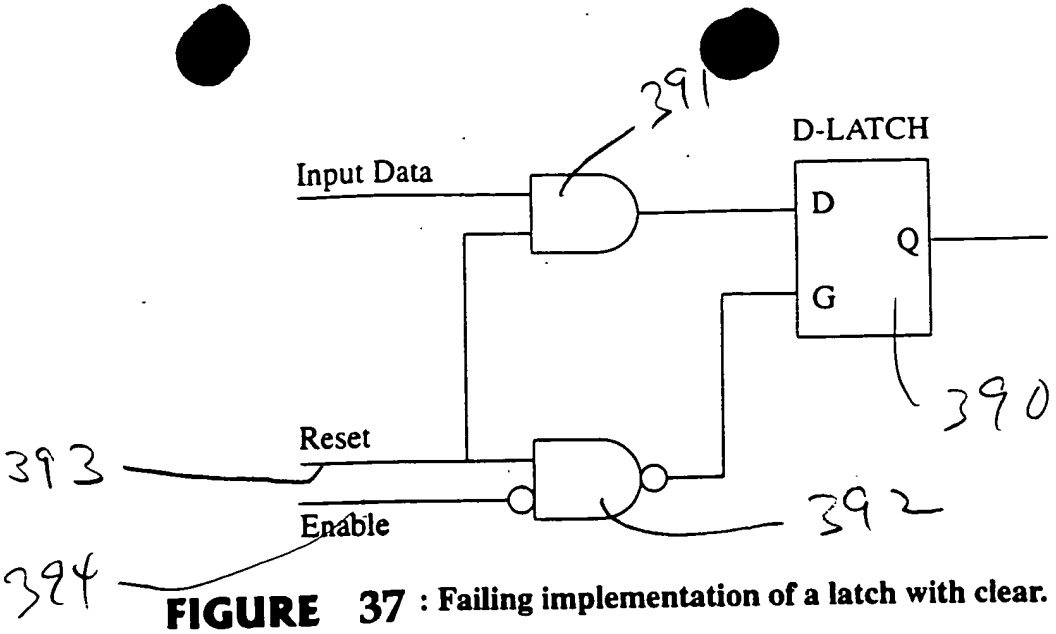


FIGURE 37 : Failing implementation of a latch with clear.

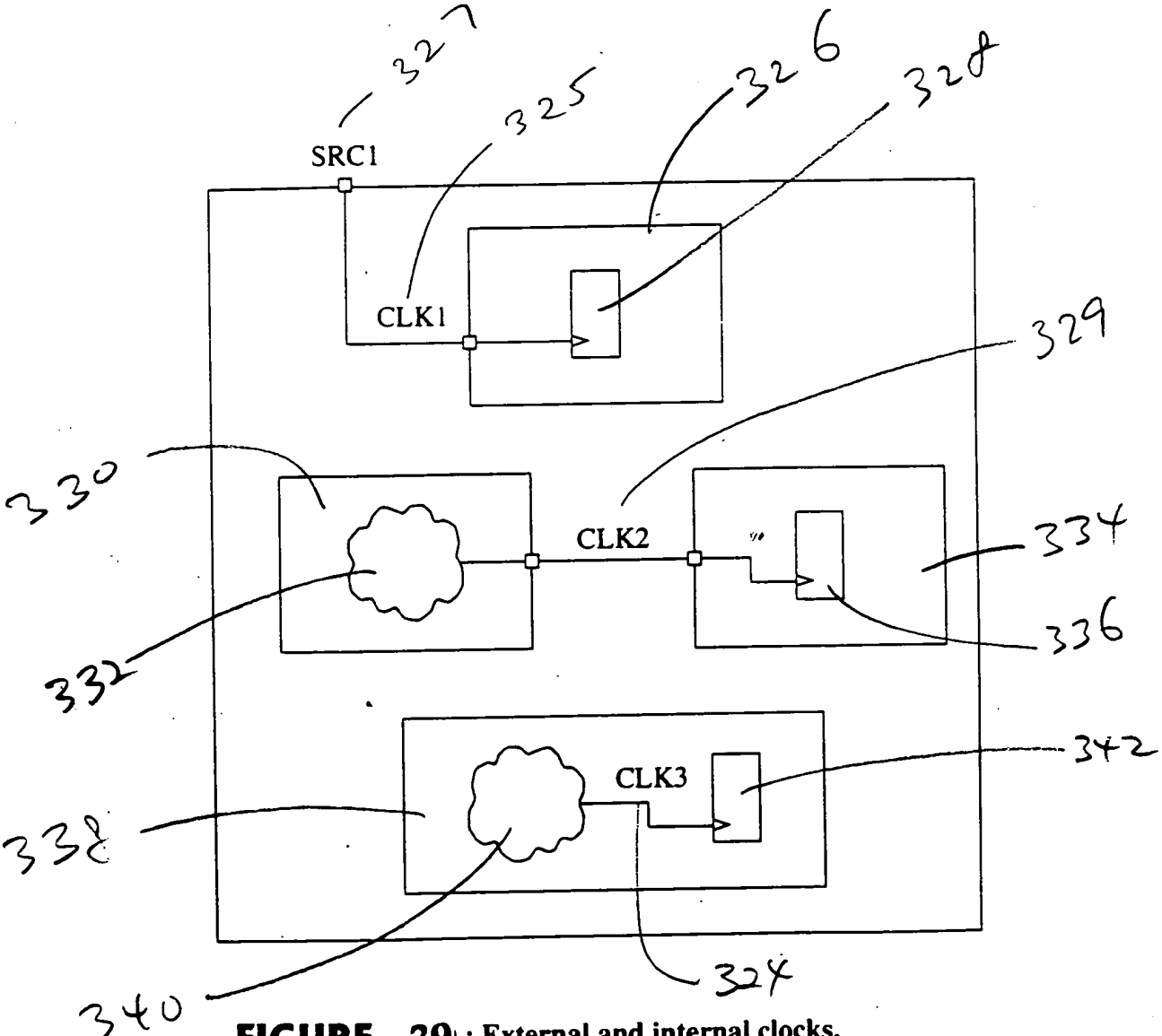


FIGURE 39 : External and internal clocks.

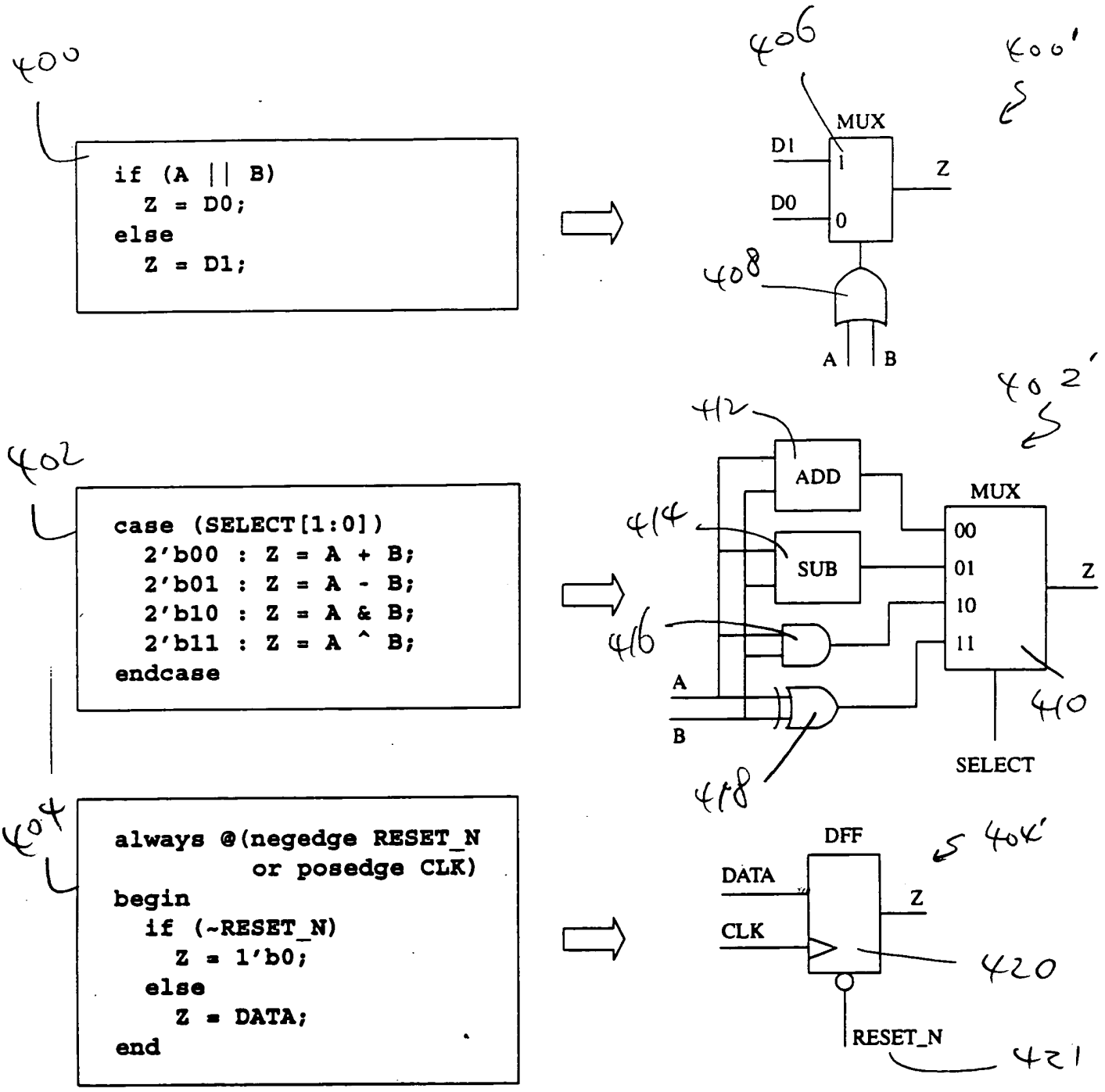


FIGURE 38 : Examples of transforms used for RTL code translation.

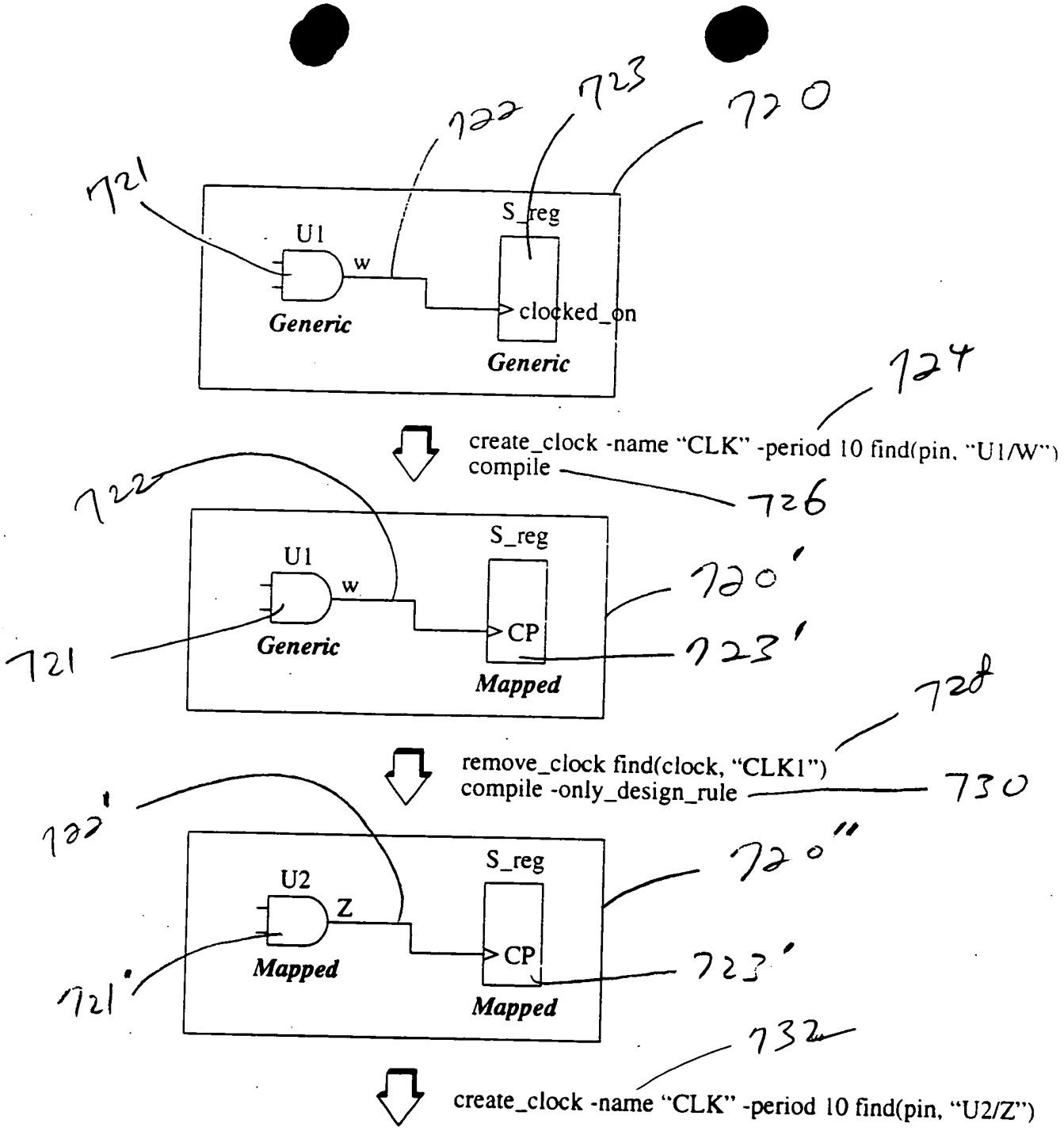


FIGURE 40 : **Process used to map cells that create internal clocks.**

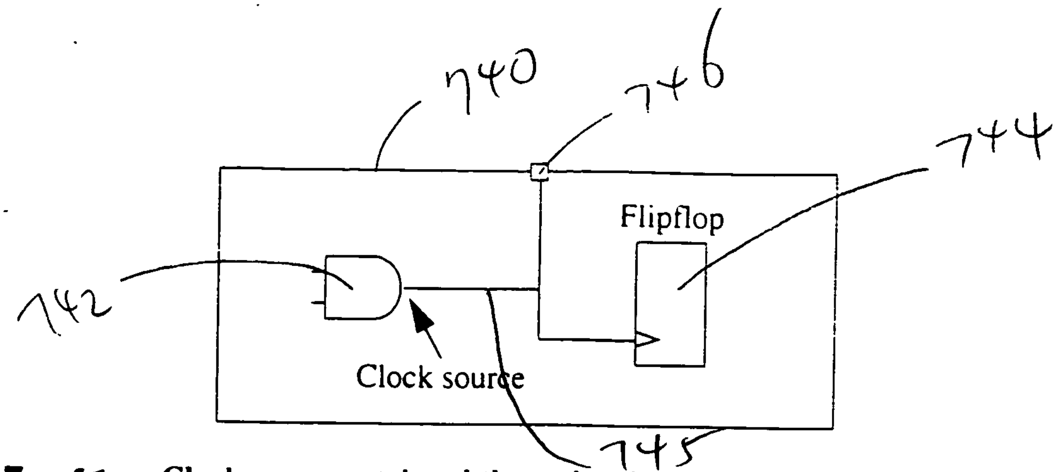


FIGURE 41A Clock source retrieved through using a connected port.

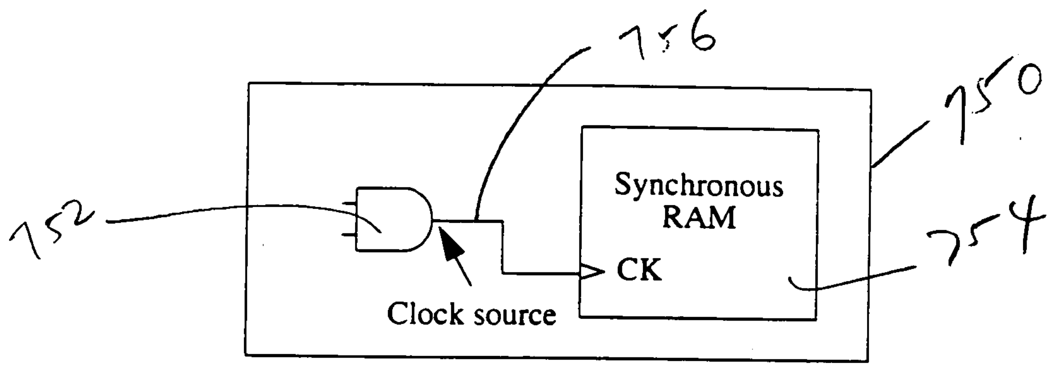


FIGURE 41B Clock retrieved through using a connected clock input pin on a RAM.

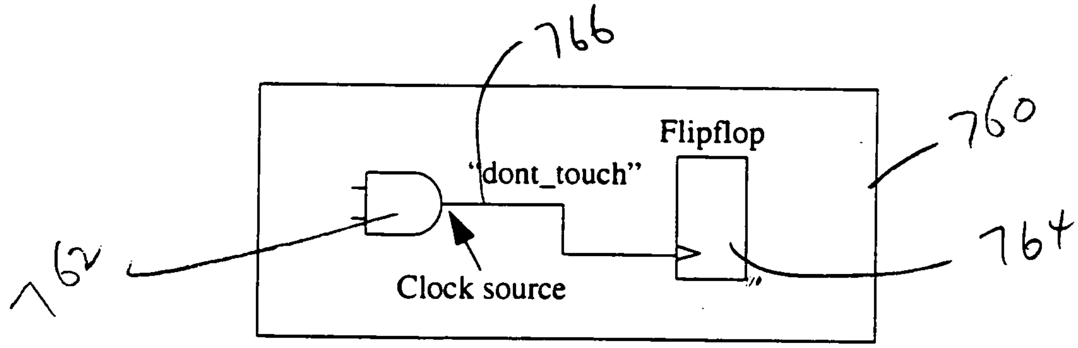


FIGURE 41C Clock source retreived through using the connected net.

: Retrieving names of new source pins.

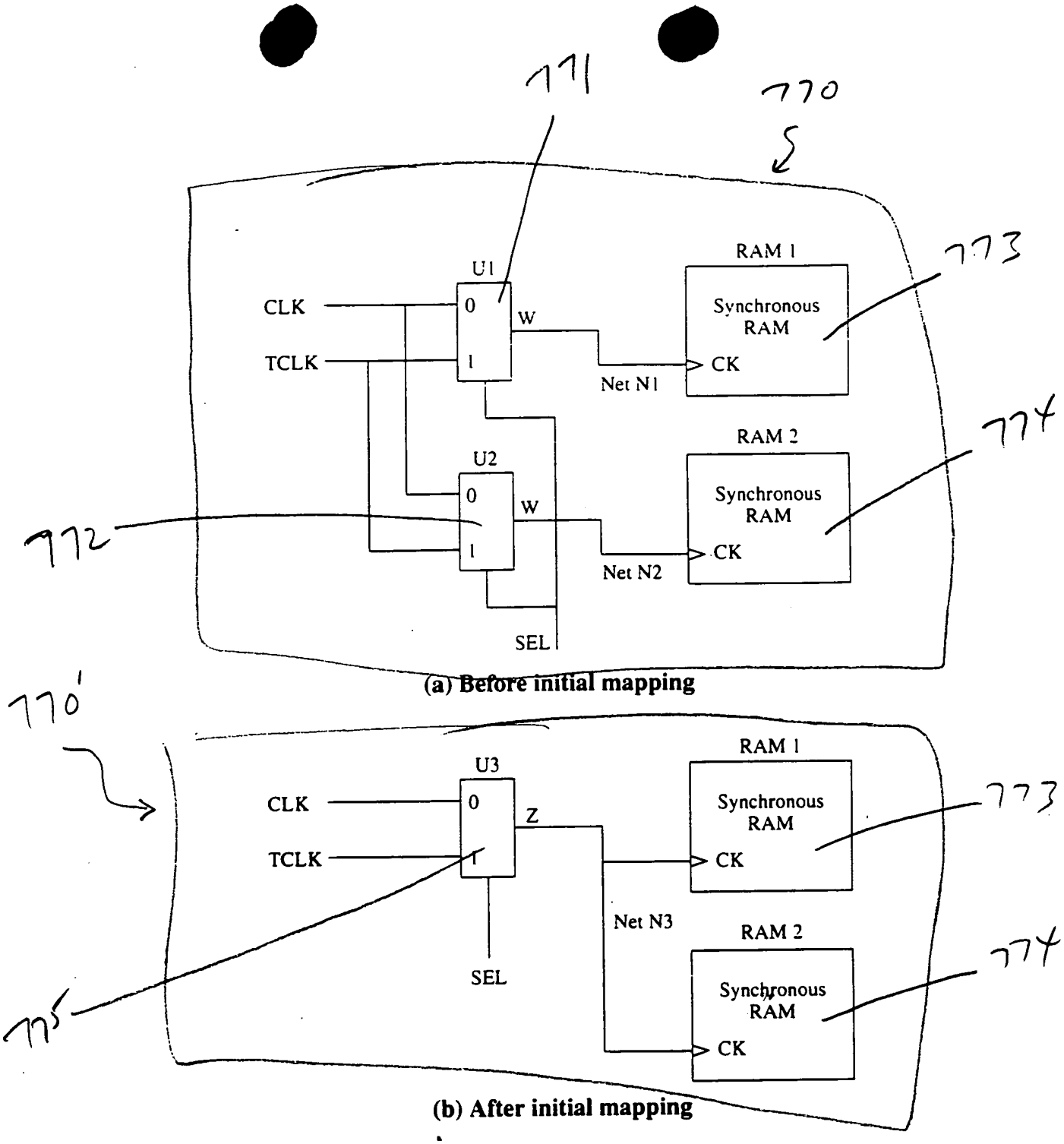
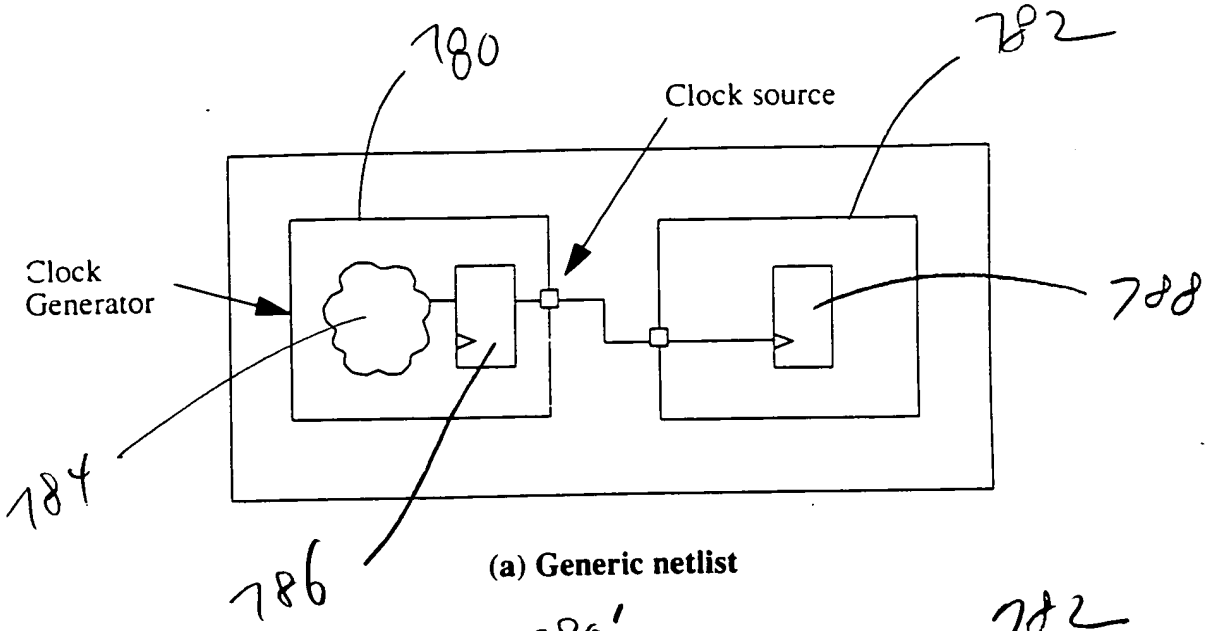
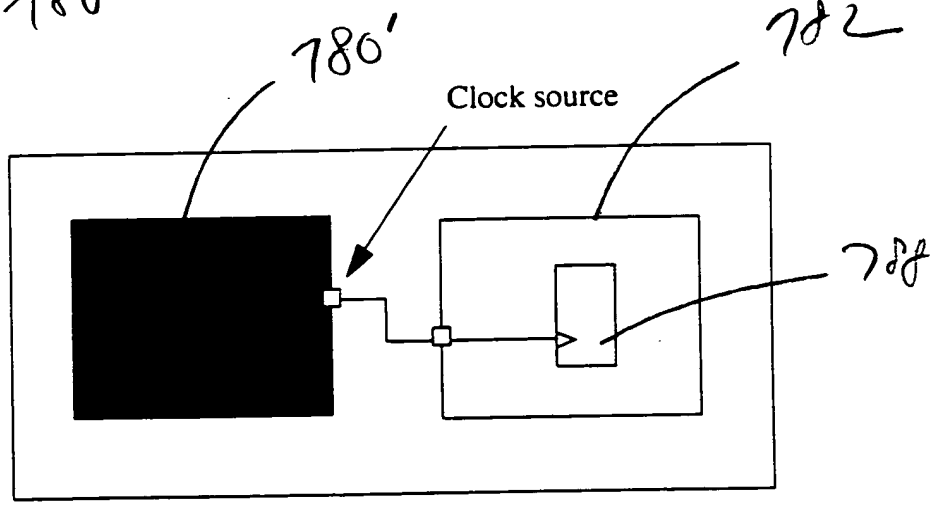


FIGURE 42 : Example of internal clocks altered through initial mapping.



(a) Generic netlist



(a) After making the clock generator a blackbox for VEGA analysis

FIGURE 43 : Handling clock generators with a "backbox_design" directive.

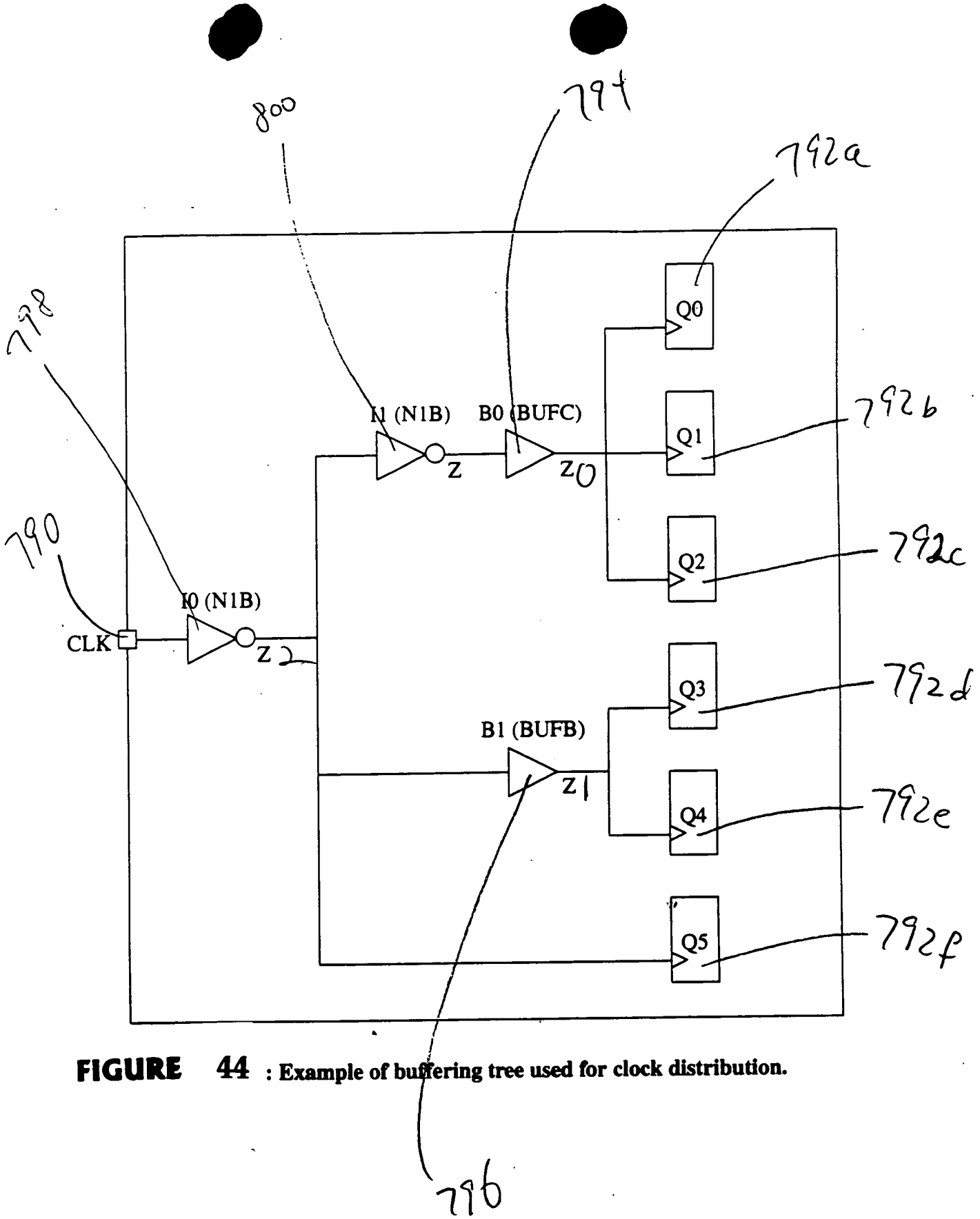


FIGURE 44 : Example of buffering tree used for clock distribution.

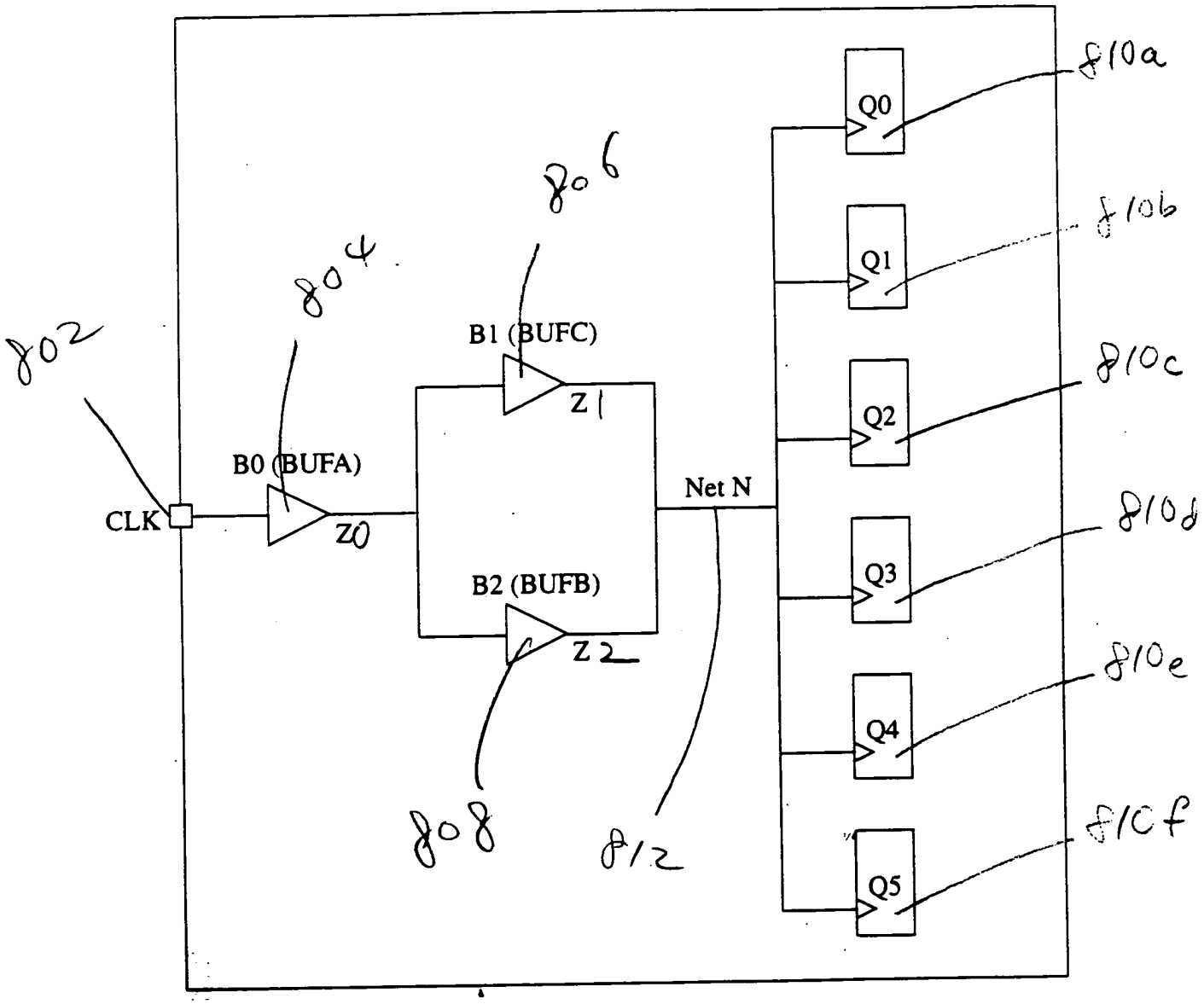


FIGURE 45 : Example of parallel buffers.

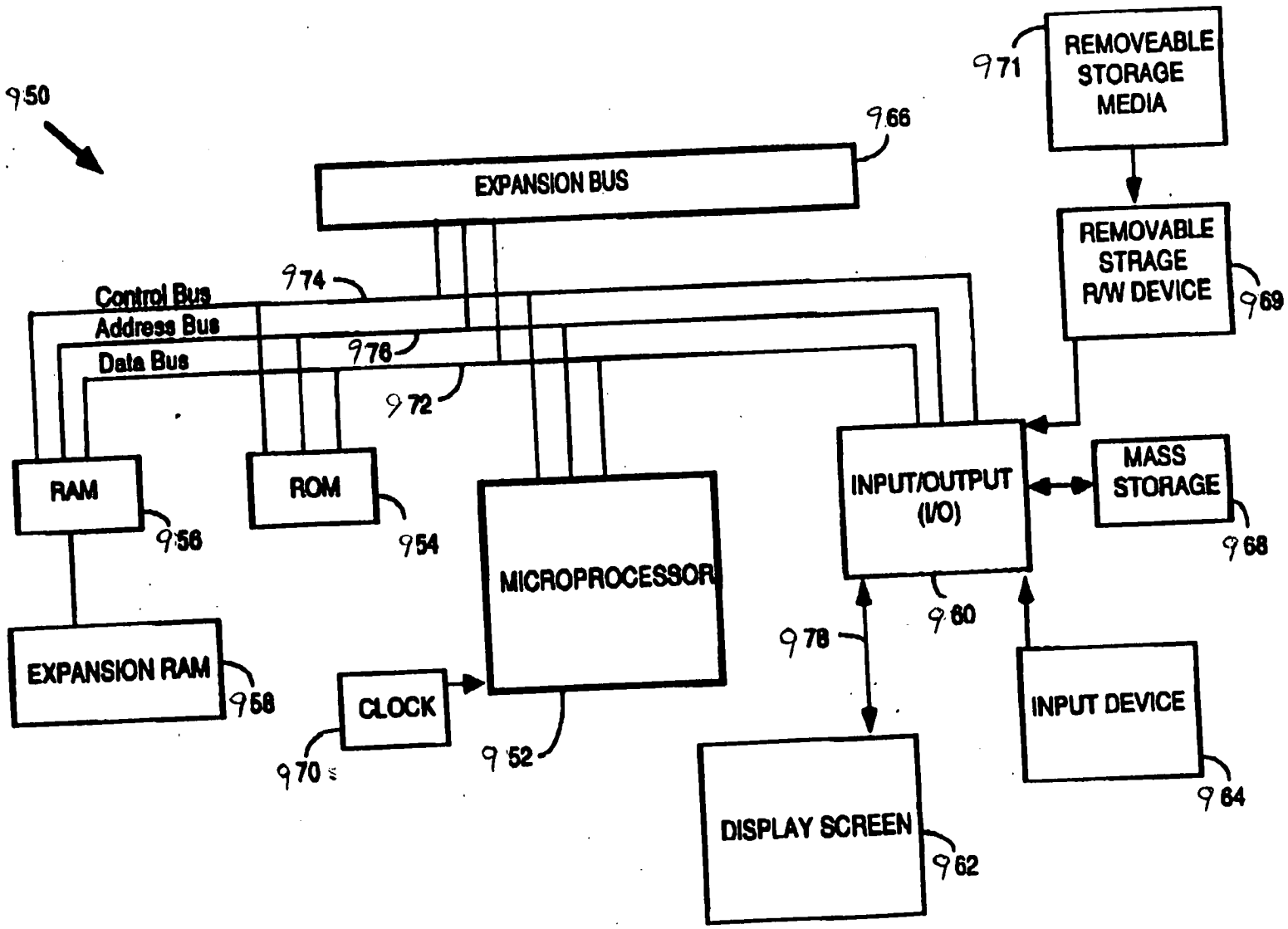

950
966
EXPANSION BUS
971
REMOVEABLE STORAGE MEDIA
REMOVABLE STRAGE R/W DEVICE
969
974
Control Bus
Address Bus
Data Bus
976
972
RAM
956
ROM
954
MICROPROCESSOR
INPUT/OUTPUT (I/O)
960
MASS STORAGE
968
978
EXPANSION RAM
958
CLOCK
970
952
INPUT DEVICE
964
DISPLAY SCREEN
962

Fig 46

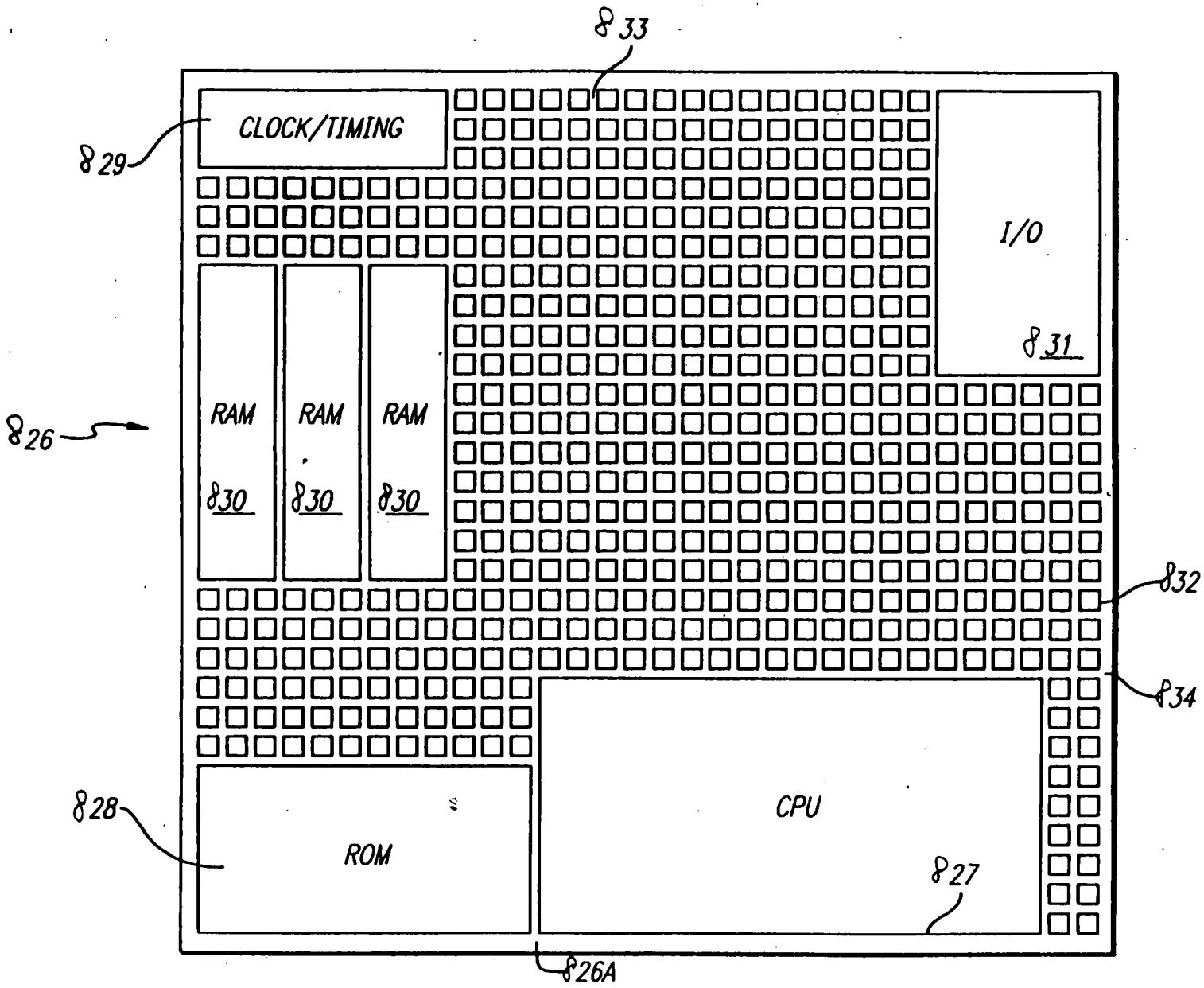

833
829
CLOCK/TIMING
I/O
831
826
RAM
830
RAM
830
RAM
830
832
834
828
ROM
CPU
827
826A

FIG. 47